1999

Using the Frechet derivative to improve Arnoldi's method

Huai-An Sun
San Jose State University

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI®

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

USING THE FRÉCHET DERIVATIVE TO IMPROVE ARNOLDI'S METHOD

A Thesis

Presented to

The Faculty of the Department of Mathematics

San Jose State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

Huai-An Sun

August. 1999

APPROVED FOR THE DEPARTMENT OF MATHEMATICS

Mohammad Saleem

Dr. Mohammad Saleem

Dr. Ho Kuen Ng

Dr. Richard Pfiefer

APPROVED FOR THE UNIVERSITY

William Fisher

ABSTRACT

USING THE FRÉCHET DERIVATIVE TO
IMPROVE ARNOLDI'S METHOD

by
Huai - An Sun

One recently developed Krylov method for solving linear systems is Arnoldi's method. Arnoldi's method is an orthogonal projection method onto a Krylov subspace $\mathbf{K}_m$ for general non-Hermitian matrices [16]. In this thesis, the method is applied to ill-conditioned and non-symmetric matrices. The results in this thesis suggest that the choice of the initial guess plays an important role in deciding how Arnoldi's method will converge or when Arnoldi's method will converge by using the Fréchet derivative. In this thesis, the Fréchet derivative is a tool to find the best possible initial guess of linear systems of equations for Arnoldi's method. Also, the thesis compares convergence with 3 different algorithms including Arnoldi's method, Arnoldi's method improved by the Fréchet derivative, and Matlab's "backslash" method.

ACKNOWLEDGMENT

I would like to thank Dr. Saleem for tirelessly working with me towards this thesis. I appreciate his patience and helpful comments in seeing me through this endeavor. I would also like to thank Dr. Ng and Dr. Pfiefer for their suggestions and contributions toward this thesis. With their help, this thesis has become a sound document.

CONTENTS

INTRODUCTION

In a practical large-scale engineering computation of the mid-1990s, where iterative algorithms are successful, perhaps a typical result is that they beat direct algorithms such as Gaussian elimination and Householder Triangulation by a factor on the order of 10 [6]. Most of the existing practical iterative techniques for solving large linear systems of equations utilize in one way or another a projection process [20]. One of the most popular iterative methods is called Arnoldi's method which is an orthogonal projection method onto a Krylov subspace $\mathbf{K}_m$ for general non-Hermitian matrices. The procedure was introduced in 1951 as a means of reducing a dense matrix into Hessenberg form [20], [21].

This thesis applies Arnoldi's method to solve systems of equations $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is an nXn matrix, $\mathbf{b}$ is a known and nonzero vector, and $\mathbf{x}$ is an unknown vector. Based on the Arnoldi process, choosing a good initial vector $\mathbf{x}_0$ plays an important role. In order to facilitate convergence, the Fréchet derivative is used as a tool to find the best possible approximations for the exact solution of $\mathbf{Ax} = \mathbf{b}$.

The Fréchet derivative, a bounded linear operator, builds the necessary implements to construct the required local linearizations. This derivative finds a linear approximation to a nonlinear operator in a neighborhood of some given point (a local linearization) [13]. By the Mean Value Theorem in which the ordinary derivative is replaced by the Fréchet derivative, the asymptotic rate of convergence is determined by the Fréchet derivative.

This thesis applies Arnoldi's method to a system of equations involving the Hilbert matrix, the Wilkinson matrix, a Toeplitz matrix, and certain random matrices. Three different algorithms, Matlab's "backslash", Arnoldi, and Arnoldi improved by the Fréchet derivative are used on these matrices.

Chapter 1 introduces the basic theory of Arnoldi's method, derives the method, and develops the algorithm. In Chapter 2, the Fréchet derivative is defined in one, two and n-dimensions. Examples of calculating the Fréchet derivative and properties of the Fréchet derivative are presented. In Chapter 3, improvements to Arnoldi's method are proposed and modifications to the algorithm for using the Fréchet derivative are presented. Chapter 4, which contains the numerical results, examines how the Fréchet derivative can improve the efficiency of Arnoldi's method for ill-conditioned/symmetric systems and how ineffective it is for a non-symmetric system. The matrices studied are 5x5, 15x15, 20x20, 30x30, and 50x50. Chapter 5 summarizes all the observations and discusses possibilities for future research.

CHAPTER 1.

ARNOLDI'S METHOD

Section 1.1 Projection Method.

Consider solving the linear system

$$\mathbf{Ax} = \mathbf{b} \tag{1.1}$$

where $\mathbf{A}$ is an $n \times n$ real matrix. The idea of projection techniques is to extract an approximate solution to the above problem from an m-dimensional subspace of R^n. If $\mathbf{K}$ is this subspace of candidate approximates, or search subspace of R^n, and if m is its dimension, then, in general, m constraints must be imposed on $\mathbf{K}$ to be able to extract such an approximation. A typical way of describing these constraints is to impose m independent orthogonality conditions. Let $\mathbf{L}$ be the m-dimensional subspace of R^n which is required to be orthogonal to the residual vector $\mathbf{r} = \mathbf{b} - \mathbf{Ax}$. This simple framework is common to many different projection methods and is called the *Petrov-Galerkin conditions* [6], [20].

A projection technique onto the subspace $\mathbf{K}$ and orthogonal to $\mathbf{L}$ is a process that finds an approximate solution $\tilde{\mathbf{x}}$ of (1.1) by imposing the condition that the residual vector $\mathbf{r}$ be orthogonal to $\mathbf{L}$: find $\tilde{\mathbf{x}} \in \mathbf{K}$ such that $\mathbf{b} - \mathbf{A}\tilde{\mathbf{x}} \perp \mathbf{L}$. The process should be repeated until n linearly independent residuals are obtained. If the residual is zero, then there is an exact solution [6]. However, in practice, due to round-off errors whenever a computer is used to perform real-number calculations, an exact solution can not always be found. If $\tilde{\mathbf{x}} = \mathbf{x}_0 + \mathbf{z}$ is the approximate solution of (1.1) where $\mathbf{x}_0$ is an initial guess

and $r_0 = b - Ax_0$ is the initial residual, then

$$b - A\tilde{x} = b - A(x_0 + z) = r_0 - Az$$

Therefore, the approximate problem can be defined as

$$\tilde{x} = x_0 + z,\ z \in K$$

$$r_0 - Az \perp L$$

The orthogonality condition imposed on the new residual $r = r_0 - Az$ is illustrated in Figure 1.1.1

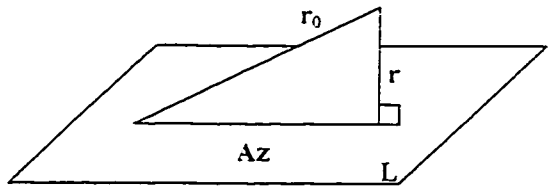

Figure 1.1.1 Intepretation of the orthogonality condtion

Let $V_m = [v_1, v_2, \ldots\ldots, v_m]$ be a system of m linearly independent vectors in R^n. The projection process onto the subspace K_m = span $(v_1, v_2, \ldots\ldots, v_m)$ seeks an approximation x_m to the solution of (1.1) by requiring that

$$x_m \in K_m \text{ where } x_m = x_0 + z_m$$

$$b - Ax_m \perp v_j, \quad j = 1, 2, \ldots\ldots\ldots, m \qquad (1.2)$$

Writing $x_m = V_m \cdot y_m$, where (* . *) denotes the inner product, V_m is an $n \times m$ matrix with column vector $v_1, v_2, \ldots, v_m$, and y_m are column vector. According to (1.2), y_m must satisfy the $m \times m$ linear system by requiring that

$V_m^T(b - Ax_m) = 0$. Then it is obviously

$$\mathbf{V_m}^T\mathbf{b} - \mathbf{V_m}^T\mathbf{A}\mathbf{V_m} \cdot y_m = 0,$$

where $\mathbf{V_m}^T$ denotes the transpose of $\mathbf{V_m}$. If $\mathbf{Q_m}$ denotes the orthogonal projection onto the subspace $\mathbf{K_m}$, then, (1.2) can be written as

$$\mathbf{x_m} \in \mathbf{K_m}$$

$$\mathbf{Q_m}(\mathbf{b} - \mathbf{A}\mathbf{x_m}) = 0$$

For simplicity, assume that $\mathbf{b} \in \mathbf{K_m}$ and $\mathbf{A_m}$ is the restriction of $\mathbf{Q_m}\mathbf{A}$ to $\mathbf{K_m}$, so that $\mathbf{x_m}$ is the solution in $\mathbf{K_m}$ for this equation

$$\mathbf{b} - \mathbf{A_m}\mathbf{x} = \mathbf{0} \tag{1.3}$$

(Note that $\mathbf{b} \in \mathbf{K_m}$, so that $\mathbf{Q_m}\mathbf{b} = \mathbf{b}$) [22].

The equation (1.1) is therefore replaced by the m-dimensional problem (1.3). This is a basic projection step in its most general form. Projection methods as described above form a unifying framework for most of the iterative techniques [6], [16].

In order to study the convergence properties of this process, one may express the error in terms of the distance between the exact solution $\mathbf{x}^*$ and the subspace $\mathbf{K_m}$, that is in terms of $\|(\mathbf{I} - \mathbf{Q_m})\mathbf{x}^*\|$ (see Figure 1.1.2) [20], [22].

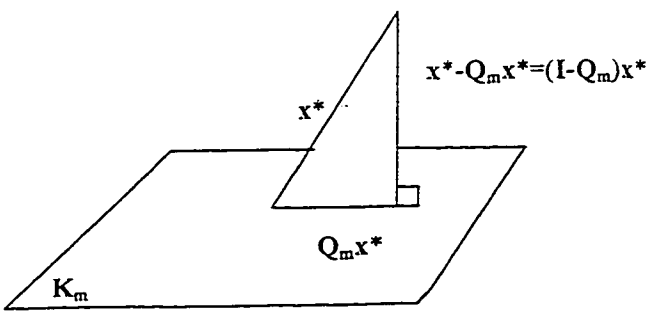

Figure 1.1.2 Orthogonai projector

When A is Hermitian positive definite, the convergence is more easily studied by using the fact that the approximate solution $\mathbf{x}_m$ minimizes the error function

$$\mathbf{E}(\mathbf{x}) = (\mathbf{x} - \mathbf{x}^*)^T \mathbf{A}(\mathbf{x} - \mathbf{x}^*)$$

over all elements $\mathbf{x}$ in $\mathbf{K}_m$ [22]. Unfortunately, this property does not extend to the nonsymmetric case, so it becomes necessary to make a different approach. Suppose that the exact solution $\mathbf{x}^*$ is close to $\mathbf{K}_m$, in that $\mathbf{Q}_m \mathbf{x}^*$ is close to $\mathbf{x}^*$. Then it is possible to show that $\mathbf{x}_m$ is close to $\mathbf{x}^*$ by showing that the residual of $\mathbf{Q}_m\mathbf{x}^*$ for the equation (1.3) is small. More precisely,

Corollary 1.1.1: Let $\gamma_m = \| \mathbf{Q}_m\mathbf{A}(\mathbf{I} - \mathbf{Q}_m) \|$. Then the residual of $\mathbf{Q}_m\mathbf{x}^*$ got (1.3) satisfies

$$\| \mathbf{b} - \mathbf{A}_m\mathbf{Q}_m\mathbf{x}^* \| \leq \gamma_m \| (\mathbf{I} - \mathbf{Q}_m)\mathbf{x}^* \|$$

(see proof [20], [22]).

As a consequence, the next corollary gives a bound for $\| \mathbf{x}^* - \mathbf{x}_m \|$.

Corollary 1.1.2: Let γ_m be defined as above and let k_m be the norm of the inverse of $\mathbf{A}_m$. Then the error $\mathbf{x}^* - \mathbf{x}_m$ satisfies

$$\| \mathbf{x}^* - \mathbf{x}_m \| \leq \sqrt{1 + r_m^2 k_m^2}\ \| (\mathbf{I} - \mathbf{Q}_m)\mathbf{x}^* \|$$

(see proof [22]).

Section 1.2 Krylov Subspace Method.

As seen in the previous section, a general projection method for solving the linear system $\mathbf{Ax} = \mathbf{b}$ is a method which seeks an approximate solution $\mathbf{x}_m$ from an affine subspace $\mathbf{x}_0 + \mathbf{K}_m$ of dimension m by imposing the Petrov-Galerkin condition

$$\mathbf{b} - \mathbf{Ax}_m \perp \mathbf{L}_m,$$

where $\mathbf{L}_m$ is another subspace of dimension m and is required to be orthogonal to the residual vector (see Figure 1.2.1). Here $\mathbf{x}_0$ represents an arbitrary initial guess to the exact solution $\mathbf{x}^*$. A Krylov subspace method is a method in which the subspace $\mathbf{K}_m$ is the Krylov subspace:

$$\mathbf{K}_m(\mathbf{A}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, \mathbf{Ar}_0, \mathbf{A}^2\mathbf{r}_0, \ldots\ldots, \mathbf{A}^{m-1}\mathbf{r}_0\}, \tag{1.4}$$

where $\mathbf{r}_0 = \mathbf{b} - \mathbf{Ax}_0$. When there is no ambiguity, $\mathbf{K}_m(\mathbf{A}, \mathbf{r}_0)$ is denoted by $\mathbf{K}_m$ [4], [11]. If the unknown $\mathbf{x}$ decomposed as $\mathbf{x} = \mathbf{x}_0 + \mathbf{z}$, then clearly the new unknown $\mathbf{z}$ must satisfy

$$\begin{aligned} \mathbf{Az} - \mathbf{r}_0 &= \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{z}) \\ &= \mathbf{b} - \mathbf{Ax} = 0. \end{aligned} \tag{1.5}$$

By a Krylov subspace method, an approximation $\mathbf{z}_m$ is obtained to the equation (1.5) by applying a projection process to the system (1.5) onto the Krylov subspace $\mathbf{K}_m = \text{span}\{\mathbf{r}_0, \mathbf{Ar}_0, \ldots\ldots, \mathbf{A}^{m-1}\mathbf{r}_0\}$. Assume that the vectors $\mathbf{r}_0, \mathbf{Ar}_0, \ldots\ldots, \mathbf{A}^{m-1}\mathbf{r}_0$ are linearly independent, which means that $\dim(\mathbf{K}_m) = m$. If $\mathbf{V}_m = [\mathbf{v}_1, \mathbf{v}_2, \ldots\ldots, \mathbf{v}_m]$ is any basis of $\mathbf{K}_m$, then according to Section 1.1, $\mathbf{z}_m$ can be expressed as $\mathbf{z}_m = \mathbf{V}_m \cdot \mathbf{y}_m$, where $\mathbf{y}_m$ is the solution of the $m \times m$ system

$$\mathbf{V}_m^T\mathbf{r}_0 - \mathbf{V}_m^T\mathbf{AV}_m \cdot \mathbf{y}_m = 0,$$

and the approximate $\mathbf{x}_m$ of $\mathbf{Ax} = \mathbf{b}$ is related to $\mathbf{z}_m$ by $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{z}_m$. If $\mathbf{z}^* = \mathbf{A}^{-1}\mathbf{r}_0$ denotes the exact solution of the system (1.5), then

$$\mathbf{x}^* - \mathbf{x}_m = \mathbf{z}^* - \mathbf{z}_m, \tag{1.6}$$

which means that $\mathbf{x}_m$ and $\mathbf{z}_m$ admit the same error vector for (1.1) and (1.5) [22].

Although all the projection techniques provide the same approximations, the choice of $\mathbf{L}_m$ i.e., the constraints used to build these approximations will have a substantial effect on the iterative technique. For example, Arnoldi's method simply lets $\mathbf{L}_m = \mathbf{K}_m$ (see Figure 1.2.1) and the Generalized Minimum Residual method (GMRES) lets $\mathbf{L}_m = \mathbf{AK}_m$. It is known as an orthogonal projection method [20]. In the next section, Arnoldi's method will be derived and its algorithm will be developed. This paper will not discuss GMRES, but Brown's paper [11] describes and compares the differences and advantages of both Arnoldi's method and GMRES.

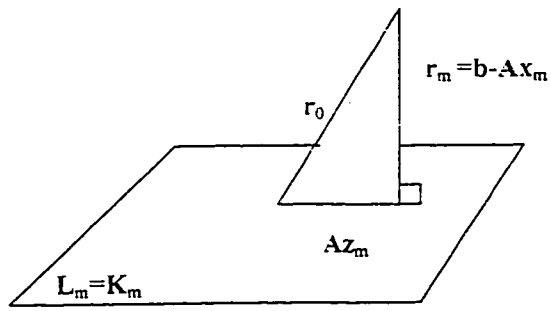

Figure 1.2.1 Arnoldi's method

Section 1.3 Arnoldi's method.

Arnoldi's method is an orthogonal projection method onto $\mathbf{K}_m$ for general non-Hermitian matrices ($\mathbf{A} = \mathbf{A}^*$). The procedure was introduced in 1951 as a means of reducing a dense matrix into Hessenberg form [6].

Consider the linear system $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A}$ is an $n \times n$ real matrix which is nonsingular, and $\mathbf{x}$ and $\mathbf{b}$ are vectors of length n. Arnoldi's method generates a sequence of iterates converging to the solution of $\mathbf{Ax} = \mathbf{b}$. Let $\mathbf{x} = \mathbf{x}_0 + \mathbf{z}$ where $\mathbf{x}_0$ is an initial guess for the exact solution of (1.1). Then,

$$0 = \mathbf{b} - \mathbf{Ax} = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{z}) = \mathbf{r}_0 - \mathbf{Az},$$

where $\mathbf{r}_0 = \mathbf{b} - \mathbf{Ax}_0$ is the initial residual. Let $\mathbf{K}_m$ be the Krylov subspace of (1.4). Arnoldi's method finds an approximate solution $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{z}_m$ such that

$$(\mathbf{b} - \mathbf{Ax}_m) \perp \mathbf{K}_m \text{ (equivalently } (\mathbf{r}_0 - \mathbf{Az}_m) \perp \mathbf{K}_m). \tag{1.7}$$

Note that: $\mathbf{r}_m = \mathbf{b} - \mathbf{Ax}_m = (\mathbf{r}_0 + \mathbf{Ax}_0) - \mathbf{A}(\mathbf{x}_0 + \mathbf{z}_m) = \mathbf{r}_0 - \mathbf{Az}_m$

Arnoldi's method starts by generating orthonormal vector $\mathbf{v}_i$, i = 1,....., m, and then builds the vector $\mathbf{x}_m$ that satisfies (1.7). In Arnoldi's method, the $\mathbf{v}_i$'s are computed such that they form an orthonormal basis of the Krylov subspace $\mathbf{K}_m$, where the first vector $\mathbf{v}_1$ is obtained by normalizing $\mathbf{r}_0$. The details of the computation follow:

Suppose that $\mathbf{x}$ is the exact solution for $\mathbf{Ax} = \mathbf{b}$. Let $\mathbf{x}_0$ be an initial vector of $\mathbf{x}$, then

$\mathbf{r}_0 = \mathbf{b} - \mathbf{Ax}_0$ will be the initial residual.

Construct the first iteration $\mathbf{x}_1$ and the residual $\mathbf{r}_1$ as follows:

Let $\mathbf{x}_1 = \mathbf{x}_0 + \mathbf{z}_1$ where $\mathbf{z}_1 \in \mathbf{K}_1 = \text{span}\{\mathbf{r}_0\} = \text{span}\{\mathbf{v}_1\}$(i.e. there exists scalars such that

$z_1 = sr_0$) such that $r_1 = b - Ax_1 = r_0 - Az_1$ is orthogonal to r_0. In other words, r_1 is orthogonal K_1. And $r_1 \in K_2 = \text{span}\{r_1, Ar_0\} = \text{span}\{r_0, r_1\} = \text{span}\{v_1, v_2\}$ where $v_2 = \pm r_1/\|r_1\|$. Note that v_2 is orthogonal to v_1.

Construct the second iteration x_2 and the residual r_2 as follows:

Let $x_2 = x_0 + z_2$, where $z_2 \in K_2 = \text{span}\{r_0, Ar_0\}$, so there exists scalars s and t such that $z_2 = sr_0 + tAr_0$. Require that $r_2 = b - Ax_2 = r_0 - Az_2$ be orthogonal to r_0 and r_1.

In other word, r_2 is orthogonal to K_2. Let $v_3 = = \pm r_2/\|r_2\|$. So v_3 is orthogonal to v_1 and v_2. Then $r_2 \in K_3 = \text{span}\{r_0, Ar_0, A^2r_0\} = \text{span}\{r_0, r_1, r_2\} = \text{span}\{v_1, v_2, v_3\}$

Proceed inductively at mth iteration:

Construct x_m and the corresponding residuals r_m, where m = 3, 4,...n, by discovering $x_m = x_0 + z_m$, where $z_m \in K_m = \text{span}\{r_0, Ar_0, A^2r_0, ..., A^{m-1}r_0\}$,

such that $r_m = b - Ax_m = r_0 - Az_m$ is orthogonal to r_0, r_1,....., and r_{m-1}. In other words, r_m is orthogonal to $K_{m-1} = \text{span}\{r_0, Ar_0, A^2r_0, ..., A^mr_0\}$.

The computational method is to select some m and use the Gram-Schmidt Process to set $\{r_0, Ar_0, A^2r_0, ..., A^{m-1}r_0\}$ to generate orthonormal vectors v_i which will be in the direction of r_{i-1} where i = 1, 2,....., m, the procedure is as follows:

Let x_0 be an initial guess for x and $r_0 = b - Ax_0$ be the initial residual.

Let $\beta = \|r_0\|$. Suppose

$$K_{m-1} = \text{span}\{r_0, Ar_0, A^2r_0, ..., A^{m-2}r_0\} = \text{span}\{v_1, v_2, ..., v_{m-1}\},$$

where $\{v_1, v_2, ..., v_{m-1}\}$ is an orthonormal basis for K_{m-1}. Proceed inductively.

Let $\mathbf{v}_2 = \mathbf{A}\mathbf{v}_1$. First choose $\mathbf{v}_1 = \mathbf{r}_0/\beta$.

Construct $\mathbf{v}_2$ as follows:

$h_{11} = (\mathbf{A}\mathbf{v}_1, \mathbf{v}_1)$ where (* , *) denotes inner product

Let $\mathbf{w}_1$ be $\mathbf{v}_2$ minus projection of $\mathbf{v}_2$ on $\mathbf{v}_1$, so

$\mathbf{w}_1 = \mathbf{v}_2 - h_{11}\mathbf{v}_1 = \mathbf{v}_2 - (\mathbf{v}_2 , \mathbf{v}_1)\mathbf{v}_1$.

Let $h_{21} = \|\mathbf{w}_1\|$ and $\mathbf{v}_2 = \mathbf{w}_1/h_{21} = \mathbf{w}_1/\|\mathbf{w}_1\|$.

Construct $\mathbf{v}_3$ as follow:

Let $\mathbf{v}_3 = \mathbf{A}\mathbf{v}_2 = \mathbf{A}(\mathbf{A}\mathbf{v}_1) = \mathbf{A}^2\mathbf{v}_1$.

$h_{12} = (\mathbf{A}\mathbf{v}_2, \mathbf{v}_1)$ and $h_{22} = (\mathbf{A}\mathbf{v}_2, \mathbf{v}_2)$

Let $\mathbf{w}_2$ be $\mathbf{v}_3$ minus projection of $\mathbf{v}_3$ on $\mathbf{v}_1$ minus projection of $\mathbf{v}_3$ on $\mathbf{v}_2$, so

$\mathbf{w}_2 = \mathbf{A}\mathbf{v}_2 - h_{12}\mathbf{v}_1 - h_{22}\mathbf{v}_2 = \mathbf{v}_3 - (\mathbf{v}_3, \mathbf{v}_1)\mathbf{v}_1 - (\mathbf{v}_3, \mathbf{v}_2)\mathbf{v}_2$.

Let $h_{32} = \|\mathbf{w}_2\|$ and $\mathbf{v}_3 = \mathbf{w}_2 / h_{32} = \mathbf{w}_2/\|\mathbf{w}_2\|$.

Construct $\mathbf{v}_m$, the orthogonal basis as follow:

$$h_{i,j} = (\mathbf{A}\mathbf{v}_j, \mathbf{v}_i), \qquad i \le j+1, \tag{1.8}$$

$$\mathbf{w}_{m-1} = \mathbf{A}\mathbf{v}_{m-1} - \sum_{i=1}^{m-1} h_{i,\,m-1}\mathbf{v}_i, \tag{1.9}$$

$$h_{m,\,m-1} = \|\mathbf{w}_{m-1}\| \text{ and } \mathbf{v}_m = \mathbf{w}_{m-1}/h_{m,\,m-1} = \mathbf{w}_{m-1}/\|\mathbf{w}_{m-1}\|. \tag{1.10}$$

The process will stop if the vector $\mathbf{w}_{m-1}$ vanishes.

Note that since $\mathbf{r}_{m-1} = \mathbf{b} - \mathbf{A}\mathbf{x}_{m-1} = \mathbf{r}_0 - \mathbf{A}\mathbf{z}_{m-1}$

where $\mathbf{z}_{m-1} \in \mathbf{K}_{m-1} = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{m-2}\mathbf{r}_0\}$ and $\mathbf{r}_{m-1} \in \mathbf{K}_m = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots\ldots, \mathbf{A}^{m-1}\mathbf{r}_0\}$.

Now $\mathbf{r}_{m-1}$ is constructed to be orthogonal to $\mathbf{r}_0, \ldots, \mathbf{r}_{m-2}$ and $\mathbf{r}_{m-2} \in \mathbf{K}_{m-1}$, so $\mathbf{r}_{m-1}$ is orthogonal

to K_{m-1}. Also v_m has the same property, hence $\pm r_{m-1}/\|r_{m-1}\| = w_m/\|w_{m-1}\| = v_m$. So, r_{m-1} is orthogonal to $v_1, v_2, \ldots, v_{m-1}$ and v_m in the direction of r_{m-1}.

Proposition 1.3.1: Let V_m be the $n \times m$ matrix with column vectors $v_1, v_2, \ldots, v_m$, $\tilde{H}_m$ be $(m+1) \times m$ Hessenberg matrix and H_m be the matrix obtained from $\tilde{H}_m$ deleting its last row. Both matrices H_m and $\tilde{H}_m$ are shown in Appendix B. Let $e_i^T = [0, 0\ldots, 1, 0, \ldots, 0]$ with 1 in i^{th} position. Then

$$AV_m = V_m H_m + w_m e_m^T \tag{1.11}$$

$$= V_{m+1} \tilde{H}_m, \tag{1.12}$$

$$V_m^T A V_m = H_m. \tag{1.13}$$

Proof : The equation (1.11) is illustrated in Figure 1.3.1

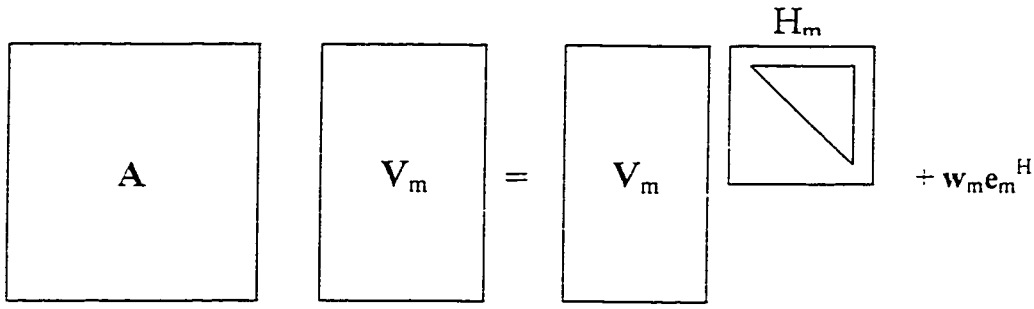

Figure 1.3.1 The action of A on V_m gives $V_m H_m$ plus a rank one matrix.

The equation of (1.12) follows from (1.9) and (1.10) :

$$Av_j = w_j + \sum_{i=1}^{j} h_{i,j} v_i = h_{j+1,j} v_{j+1} + \sum_{i=1}^{j} h_{i,j} v_i$$

$$= h_{j+1,j} v_{j+1} + \sum_{i=1}^{j} h_{i,j} v_i$$

$$= \sum_{i=1}^{j+1} h_{i,j} v_i, \qquad j = 1, 2\ldots, m,$$

$$= \mathbf{V}_{m+1}\tilde{\mathbf{H}}_m.$$

The equation of (1.13) follows by multiplying both sides of (1.11) by $\mathbf{V}_m^T$ to make use of the orthonormality of $[\mathbf{v}_1, \mathbf{v}_2 \ldots\ldots \mathbf{v}_m]$ [6].

As a result, the following proof will show $\mathbf{V}_m^T\mathbf{r}_0 = \beta\,\hat{\mathbf{e}}_1$.

Proof: Let $\mathbf{r}_m = \mathbf{b} - \mathbf{A}\mathbf{x}_m$ be the residual and $\mathbf{r}_m$ be the perpendicular to $\mathbf{K}_m$. It means for k = 1,.....m, $(\mathbf{v}_k, \mathbf{r}_m) = 0$, $\mathbf{r}_m \cdot \mathbf{V}_m = 0$ (equivalently $\mathbf{V}_m^T \cdot \mathbf{r}_m = 0$), where $\mathbf{V}_m = [\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_m]$ is any basis of $\mathbf{K}_m$.

From the previous section, $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{z}_m = \mathbf{x}_0 + \mathbf{V}_m\mathbf{y}_m$ where $\mathbf{z}_m \in \mathbf{K}_m$ Therefore, $\mathbf{x}_m - \mathbf{x}_0 = \mathbf{V}_m \cdot \mathbf{y}_m$ implied $\mathbf{A}(\mathbf{x}_m - \mathbf{x}_0) = \mathbf{A}\mathbf{V}_m\mathbf{y}_m$, then

$$\mathbf{A}\mathbf{x}_m - \mathbf{b} + \mathbf{b} - \mathbf{A}\mathbf{x}_0 = \mathbf{A}\mathbf{V}_m\mathbf{y}_m,$$

$$-\mathbf{r}_m + \mathbf{r}_0 = \mathbf{A}\mathbf{V}_m\mathbf{y}_m,$$

refer back to (1.13)

$$\mathbf{H}_m\mathbf{y}_m = \mathbf{V}_m^T\mathbf{A}\mathbf{V}_m\mathbf{y}_m = \mathbf{V}_m^T(\mathbf{r}_0 - \mathbf{r}_m) = \mathbf{V}_m^T\mathbf{r}_0$$

$$= [(\mathbf{v}_1,\mathbf{r}_0), (\mathbf{v}_2,\mathbf{r}_0), \ldots\ldots, (\mathbf{v}_m, \mathbf{r}_0)]^T{}_{1\times m}$$

$$= [(\mathbf{v}_1, \mathbf{r}_0), 0, 0, \ldots\ldots\ldots\ldots, 0]^T{}_{1\times m}$$

$$= [(\mathbf{v}_1, \beta\mathbf{v}_1), 0, 0, \ldots\ldots\ldots\ldots, 0]^T{}_{1\times m}$$

$$= (\mathbf{v}_1, \beta\mathbf{v}_1)[1, 0, 0, \ldots\ldots\ldots\ldots, 0]^T{}_{1\times m}$$

$$= \beta\,\hat{\mathbf{e}}_1 \text{ where } \hat{\mathbf{e}}_1 = (1, 0, \ldots\ldots, 0)^T,$$

so that

$$\mathbf{H}_m\mathbf{y}_m = \mathbf{V}_m^T\mathbf{r}_0 = \beta\,\hat{\mathbf{e}}_1. \qquad (1.14)$$

From (1.14), $\mathbf{y}_m = \mathbf{H}_m^{-1}(\mathbf{V}_m^T\mathbf{r}_0) = \mathbf{H}_m^{-1}(\beta\,\hat{\mathbf{e}}_1)$. Then it is obviously

$$\mathbf{x}_m = \mathbf{x}_0 + \mathbf{z}_m = \mathbf{x}_0 + \mathbf{V}_m \mathbf{y}_m$$

$$= \mathbf{x}_0 + \beta \mathbf{V}_m \mathbf{H}_m^{-1} \hat{\mathbf{e}}_1.$$

The summary of the algorithm of Arnoldi's method for linear systems is as follows : [11]

1. **Start:** Choose an initial guess $\mathbf{x}_0$ then compute $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$, and $\beta = \|\mathbf{r}_0\|$,

   and let $\mathbf{v}_1 = \mathbf{r}_0/\beta$.

2. **Arnoldi loop:**

   For j = 1, 2, ..., m do:

   (a) Form $\mathbf{A}\mathbf{v}_j$ and orthogonalize it against the previous $\mathbf{v}_1$, ..., $\mathbf{v}_j$ by the Gram-Schmidt process:

   $h_{i,j} = (\mathbf{A}\mathbf{v}_j, \mathbf{v}_i)$, $\quad i = 1,2 \ldots\ldots, j,$

   $\mathbf{w}_j = \mathbf{A}\mathbf{v}_j - \sum_{i=1}^{j} h_{i,j}\mathbf{v}_i$,

   $h_{i+1,j} = \|\mathbf{w}_{j+1}\|$, $\quad$ if $h_{i+1,j} = 0$ stop,

   $\mathbf{v}_{j+1} = \mathbf{w}_{j+1}/h_{j+1,j}$.

   (b) Compute the residual norm $\rho_j = \|\mathbf{b} - \mathbf{A}\mathbf{x}_j\|$, of the solution $\mathbf{x}_j$ that would be obtained if the procedure is stopped at this step.

   (c) If $\rho_j \leq \varepsilon$ set m = j and go to 3 where ε is error tolerance.

3. **Form approximate solution:**

   Define $\mathbf{H}_m$ to be the m × m Hessenberg matrix whose nonzero entries are the coefficients $h_{i,j}$ $1 \leq i \leq \min\{j + 1, m\}$, $1 \leq j \leq m$ and define

$\mathbf{V}_m = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m]$ compute $\mathbf{x}_m = \mathbf{x}_0 + \mathbf{z}_m$, where $\mathbf{z}_m = \beta \mathbf{V}_m \mathbf{H}_m^{-1} \hat{\mathbf{e}}_1$.

Proposition 1.3.2: The residual vector of the approximate solution $\mathbf{x}_m$ is such that

$$\mathbf{b} - \mathbf{A}\mathbf{x}_m = - h_{m+1,m} \mathbf{e}_m^T \mathbf{y}_m \mathbf{v}_{m+1}$$

and therefore $$\| \mathbf{b} - \mathbf{A}\mathbf{x}_m \| = h_{m+1,m} \left| \mathbf{e}_m^T \mathbf{y}_m \right| .$$

Proof: The residual vector of the approximate solution $\mathbf{x}_m$ is such that

$$\mathbf{b} - \mathbf{A}\mathbf{x}_m = \mathbf{b} - \mathbf{A}(\mathbf{x}_0 + \mathbf{V}_m \mathbf{y}_m) = \mathbf{r}_0 - \mathbf{A}\mathbf{V}_m \mathbf{y}_m$$

$$= \mathbf{r}_0 - (\mathbf{V}_m \mathbf{H}_m + \mathbf{w}_m \mathbf{e}_m^T)\mathbf{y}_m \text{ where } \mathbf{w}_m = h_{m+1,m} \mathbf{v}_{m+1}$$

$$= \beta \mathbf{v}_1 - \mathbf{V}_m \mathbf{H}_m \mathbf{y}_m - h_{m+1,m} \mathbf{e}_m^T \mathbf{y}_m \mathbf{v}_{m+1}.$$

By the definition of $\mathbf{y}_m$, $\mathbf{H}_m \mathbf{y}_m = \beta \hat{\mathbf{e}}_1$ and so $\beta \mathbf{v}_1 - \mathbf{V}_m \mathbf{H}_m \mathbf{y}_m = 0$ from which the result follows immediately [11].

CHAPTER 2.

The Fréchet Derivative

Section 2.1 Introduction and definition of the Fréchet derivative.

For nonlinear operator equations, the most efficient approach to the design of a successful computational method is often *local linearization.* In this section, the necessary tool to construct the required local linearizations is defined, namely the Fréchet derivative, which is a bounded linear operator. The Fréchet derivative of a nonlinear operator is a generalization of the derivative of a real-valued function. It can find the linear approximation to a nonlinear operator in the neighborhood of some given point (a local linearization). If the given nonlinear operator equation is replaced by its local linearization, then the (local linearization) methods can be used for linear operator equations to find an approximate solution to the nonlinear operator equation. Regarding this as a point in the appropriate function space, a local linearization of the original nonlinear operator equation can be founded again in a neighborhood of this new point. To iterate this process, then it is a generalization of Newton's method for solving an equation in a single real variable [13].

A complete (bounded and closed) normed space is called a Banach space. Let $F : S \subseteq B_1 \rightarrow B_2$ be an operator, mapping a subset S of a Banach space B_1 into a Banach space B_2. Let x_0 be an element of B_1 such that S contains a neighborhood of x_0. Then F is said to be *Fréchet differentiable* at x_0 if there is a continuous linear operator, $L : B_1 \rightarrow B_2$ such that : for every $\varepsilon > 0$, there is a $\delta(\varepsilon) > 0$ such that

$$\| F(x) - \{F(x_0) + L(x - x_0)\} \|_{B_2} \leq \varepsilon \| x - x_0 \|_{B_1} \tag{2.1.1}$$

whenever $x \in N_{\delta(\varepsilon)}(x_0) = \{x : \| x - x_0 \|_{B_1} \leq \delta(\varepsilon)\}$. $N_{\delta(\varepsilon)}(x_0)$ is a neighborhood which is contained in the subset S of B_1.

(2.1.1) can be written as,

$$F(x) = F(x_0) + L(x - x_0) + G(x, x_0) \tag{2.1.2}$$

where $G : B_1 \rightarrow B_2$ defined by $G(x, x_0) = F(x) - F(x_0) - L(x, x_0)$ satisfies

$$\lim_{\|x - x_0\| \rightarrow 0} \{ \| G(x, x_0) \|_{B_2} / \| x - x_0 \|_{B_1} \} = 0. \tag{2.1.3}$$

Let $F(x) - F(x_0)$ be locally linear at x_0. If a sufficiently small neighborhood of x_0 is picked, $F(x) - F(x_0)$ can be approximated arbitrarily closely (in the Banach space norms) by image $L(x - x_0)$ of the linear operator L.

If such a continuous and bounded linear operator L exists for a particular x_0 in B_1, denote it by $F'(x_0)$, the *Fréchet derivative* of F at x_0. Thus, if F is Fréchet differentiable at x_0, (2.1.2) can be written as

$$F(x) = F(x_0) + F'(x_0)(x - x_0) + G(x, x_0) \tag{2.1.4}$$

where $G(x, x_0)$ is given by (2.1.3) [12], [13].

Section 2.2 Computing the Fréchet derivative.

Let $f : [a, b] \subseteq R \rightarrow R$ be a differentiable real valued function, where the real line is denoted by R and let [a, b] be an interval in R. Let x_0 be in [a, b]. Then the Fréchet derivative of f at x_0 is the ordinary derivative of f at x_0, denoted by $f'(x_0)$. Since

$$f'(x_0) = \lim_{|x-x_0| \to 0} \frac{f(x) - f(x_0)}{x - x_0},$$

it follows that $f(x) = f(x_0) + f'(x_0)(x - x_0) + G(x, x_0)$, with

$$\lim_{|x-x_0| \to 0} \frac{|G(x, x_0)|}{|x - x_0|} = 0$$

Note that R is a Banach space with norm $\|x\| = |x|$ for x in R.

Assume $f : R^2 \rightarrow R$. Let f be differentiable at (x_0, y_0). If $\frac{\partial f}{\partial x}$ and $\frac{\partial f}{\partial y}$ exist at (x_0, y_0), and

$$\lim_{\|(x,y)-(x_0,y_0)\| \to 0} \frac{f(x, y) - f(x_0, y_0) - \left[\frac{\partial f}{\partial x}(x_0, y_0)\right](x - x_0) - \left[\frac{\partial f}{\partial x}(x_0, y_0)\right](y - y_0)}{\|(x, y) - (x_0, y_0)\|} = 0$$

can be written as

$$f(x, y) = f(x_0, y_0) + \left[\frac{\partial f}{\partial x}(x_0, y_0)\right](x - x_0) + \left[\frac{\partial f}{\partial y}(x_0, y_0)\right](y - y_0) + G(x, x_0; y, y_0)$$

with

$$\lim_{\|(x,y)-(x_0,y_0)\|\to 0} \frac{\|G(x,x_0;y,y_0)\|}{\|(x,y)-(x_0,y_0)\|} = 0$$

Example 1: Suppose that the operator $F : R^2 \to R$ is defined by $F(\mathbf{x}) = x_1^2 + x_1x_2 + x_2^2$ where $\mathbf{x} = (x_1, x_2)^T$. Then, the Fréchet derivative of F at $(x_1, x_2)^T$ is represented by the 1×2 matrix

$$[2x_1 + x_2, x_1 + 2x_2].$$

So far this section has shown how to compute the Fréchet derivative in one and two dimensions. The following will discuss how to compute the Fréchet derivative in n dimensions.

The n-dimensional Euclidean space, E^n, with norm

$$\|\mathbf{x}\|_2 = (\sum_{i=1}^{n} x_i^2)^{1/2}$$

is also a Banach space. Let $\mathbf{F} : S \subseteq E^n \to E^n$ be a mapping defined on a subset S of E^n which contains a neighborhood

$$\mathbf{N}_{\delta(\varepsilon)}(\mathbf{x}_0) = \{\mathbf{x} : \|\mathbf{x} - \mathbf{x}_0\| \le \delta(\varepsilon)\}$$

of a point $\mathbf{x}_0$ in S.

Now, consider a multivariate Taylor expansion of f about $\mathbf{x}_0$:

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}(\mathbf{x}_0) + \sum_{j=1}^{n} \frac{\partial F_i}{\partial x_j}(x_0)(x - x_0)_j +$$

$$\frac{1}{2}\sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial^2 F_i}{\partial x_j \partial x_k}(x_0)(x - x_0)_j(x - x_0)_k + \ldots\ldots, \quad i = 1, \ldots\ldots, n \quad (2.2.1.(a))$$

In vector notation, the previous equation becomes

$$\mathbf{F}(\mathbf{x}) = \mathbf{F}(\mathbf{x}_0) + \frac{\partial \mathbf{F}}{\partial \mathbf{x}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + \text{higher-order terms}$$

$$= \mathbf{F}(\mathbf{x}_0) + \frac{\partial \mathbf{F}}{\partial \mathbf{x}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + \mathbf{G}(\mathbf{x}, \mathbf{x}_0). \qquad (2.2.1.(b))$$

Now, let $\mathbf{x} = \mathbf{x}_0 + \mathbf{h}$, then (2.2.1(b)) can be written as

$$\mathbf{F}(\mathbf{x}_0+\mathbf{h}) = \mathbf{F}(\mathbf{x}_0) + \frac{\partial \mathbf{F}}{\partial \mathbf{x}}(\mathbf{x}_0)\mathbf{h} + \text{higher order terms}, \quad i = 1, \ldots\ldots, n$$

where $\frac{\partial \mathbf{F}}{\partial \mathbf{x}}$ is the Jacobian matrix [12], [19]. For $\mathbf{x}$ near $\mathbf{x}_0$ the quadratic and higher terms are very small. Also if $\mathbf{F}(\mathbf{x}_0) = 0$, then

$$\mathbf{F}(\mathbf{x}) \approx \frac{\partial \mathbf{F}}{\partial \mathbf{x}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0)$$

If $\mathbf{F}'(\mathbf{x}_0)$ exists, there is an $n \times n$ Jacobian matrix with elements

$$(F'(x_0))_{ij} = \frac{\partial F_i}{\partial x_j}(x_0), \qquad i, j = 1,2 \ldots\ldots\ldots\ldots, n$$

$$= \begin{pmatrix} \frac{\partial F_1}{\partial x_1} & \cdots\cdots\cdots & \frac{\partial F_1}{\partial x_n} \\ \cdot & & \cdot \\ \cdot & \cdots\cdots\cdots & \cdot \\ \cdot & & \cdot \\ \frac{\partial F_n}{\partial x_1} & \cdots\cdots\cdots & \frac{\partial F_n}{\partial x_n} \end{pmatrix}.$$

In vector notation,

$$\mathbf{F}(\mathbf{x}_0+\mathbf{h}) = \mathbf{F}(\mathbf{x}_0) + (\mathbf{h} \cdot \nabla)\mathbf{F}(\mathbf{x}_0) + \text{higher order terms.}$$

The limit

$$\mathbf{DF}(\mathbf{x}_0)(\mathbf{y}) = \lim_{a \to 0} \frac{\mathbf{F}(\mathbf{x}_0 + a\mathbf{y}) - \mathbf{F}(\mathbf{x}_0)}{a}, \tag{2.2.2}$$

if it exists, is called the *Gâteaux derivative* of **F** at $\mathbf{x}_0$ in the direction **y** [14].

For $\mathbf{F} : S \subseteq E^n \to E^n$ with S containing a neighborhood $\mathbf{N}_{\delta(\varepsilon)}(\mathbf{x}_0)$ of $\mathbf{x}_0$, suppose the partial derivatives $J_{ij}(x_0) = \frac{\partial F_i}{\partial x_j}(x_0)$ exist in $\mathbf{N}_{\delta(\varepsilon)}(\mathbf{x}_0)$ and are continuous at $\mathbf{x}_0$ for i, j = 1, 2, ..., n. Let $\mathbf{J}(\mathbf{x}_0)$ be the Jacobian matrix with elements $J_{ij}(x_0)$. Then, the following theorem is presented:

Theorem 2.2.1: The Fréchet derivative of **F** at $\mathbf{x}_0$, $\mathbf{F}'(\mathbf{x}_0)$ exists, and $\mathbf{F}'(\mathbf{x}_0) = \mathbf{J}(\mathbf{x}_0)$ if and only if $\mathbf{DF}(\mathbf{x}_0)(\mathbf{y})$ exists and $\mathbf{DF}(\mathbf{x}_0)(\mathbf{y}) = \mathbf{J}(\mathbf{x}_0)\mathbf{y}$ for every y in E^n (see proof [13]).

Example 2: Let D be an open subset of R^n, and assume that $\mathbf{F} : D \to R^n$ is in $C^1(D, R^n)$. Represent points in R^n by column vectors, and for $\mathbf{x} = (x_1, x_2, ...x_n)^T$ where $\mathbf{x} \in D$ set $\mathbf{F}(\mathbf{x}) = (F_1(\mathbf{x}), F_2(\mathbf{x}), ..., F_n(\mathbf{x}))^T$. If **h** is small by Taylor's theorem

$$\mathbf{F}(\mathbf{x} + \mathbf{h}) = \mathbf{F}(\mathbf{x}) + \mathbf{F}'(\mathbf{x})\mathbf{h} + \mathbf{r} \tag{2.2.3}$$

where **h**, **r** are column vectors, $\mathbf{F}'(\mathbf{x})$ is the Jacobian matrix (with components $(\frac{\partial F_i}{\partial F_j}(x))$, and $\|\mathbf{r}\| = o(\|\mathbf{h}\|)$. The Fréchet derivative is thus simply the Jacobian matrix [16].

Example 3 : Suppose that the operator $\mathbf{f} : R^3 \to R^3$ is defined by

$$\mathbf{f}(\mathbf{x}) = \begin{pmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ f_3(\mathbf{x}) \end{pmatrix} = \begin{pmatrix} x_1 + x_2 - 2x_3 \\ x_1 - 2x_2 + 100x_3 \\ 2x_1 - x_2 + x_3 \end{pmatrix}$$

where $\mathbf{x} = (x_1, x_2, x_3)^T$, $\mathbf{h} = (h_1, h_2, h_3)^T$

$$\nabla f = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial x_3} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial x_3} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} & \frac{\partial f_3}{\partial x_3} \end{pmatrix} = \begin{pmatrix} 1 & 1 & -2 \\ 1 & -2 & 100 \\ 2 & -1 & 1 \end{pmatrix}$$

Then, the Fréchet derivative of **f** at **x** is

$$(\mathbf{h} \cdot \nabla)\mathbf{f} = \begin{pmatrix} 1 & 1 & -2 \\ 1 & -2 & 100 \\ 2 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} h_1 \\ h_2 \\ h_3 \end{pmatrix} = \begin{pmatrix} h_1 + h_2 - 2h_3 \\ h_1 - 2h_2 + 100h_3 \\ 2h_1 - h_2 + h_3 \end{pmatrix}.$$

Example 4: Let E^2 be a 2-dimensional Euclidean space and $\mathbf{f} : E^2 \to E^2$ be viewed as a complex valued function of a complex variable, identifying a point (x, y) in E^2 with the complex number x + iy. Denote the components of f by u and v.

Then f(x, y) = u(x, y) + iv(x, y) corresponds to

$$\mathbf{f}(x, y) = \begin{pmatrix} u(x, y) \\ v(x, y) \end{pmatrix}$$

in column vector notation for an image of the mapping **f**. Applying Theorem 2.2.1, the Fréchet derivative $\mathbf{f}'(\mathbf{z}_0)$ exists at $\mathbf{z}_0 = x_0 + iy_0$ and is expressible as the Jacobian matrix $\mathbf{f}'(\mathbf{z}_0) = \mathbf{J}(\mathbf{z}_0)$ given by

$$\mathbf{J}(\mathbf{z}_0) = \begin{pmatrix} u_x & u_y \\ v_x & v_y \end{pmatrix}$$

where $u_x = \frac{\partial u}{\partial x}\Big|_{z_0}$, $u_y = \frac{\partial u}{\partial y}\Big|_{z_0}$, $v_x = \frac{\partial v}{\partial x}\Big|_{z_0}$, $v_y = \frac{\partial v}{\partial y}\Big|_{z_0}$ provided that for every w in E^2 the limit

$$\mathbf{Df}(\mathbf{z}_0) = \lim_{a \to 0} \frac{\mathbf{f}(\mathbf{z}_0 + a\mathbf{w}) - \mathbf{f}(\mathbf{z}_0)}{a}$$

exists and $\mathbf{Df}(\mathbf{z}_0)\mathbf{w} = \mathbf{J}(\mathbf{z}_0)\mathbf{w}$.

Now suppose the limit does exist for every $\mathbf{w} = w_1 + iw_2$ in E^2 and $\mathbf{Df}(\mathbf{z}_0)\mathbf{w} = \mathbf{J}(\mathbf{z}_0)\mathbf{w}$. Then, regarding $\mathbf{w}$ and $\mathbf{f}'(\mathbf{z}_0)\mathbf{w}$ as complex numbers, so that

$$\mathbf{f}'(\mathbf{z}_0)\mathbf{w} = \mathbf{J}(\mathbf{z}_0)\mathbf{w} = \begin{pmatrix} u_x w_1 + u_y w_2 \\ v_x w_1 + v_y w_2 \end{pmatrix}$$

$$= (u_x w_1 + u_y w_2) + i(v_x w_1 + v_y w_2)$$

$$= \frac{(u_x w_1 + u_y w_2) + i(v_x w_1 + v_y w_2)}{\mathbf{w}} \mathbf{w};$$

since $\mathbf{w} = w_1 + iw_2$, it follows

$$\mathbf{f}'(\mathbf{z}_0) = \frac{(u_x w_1 + u_y w_2 + i(v_x w_1 + v_y w_2)}{w_1 + iw_2} \cdot \frac{w_1 - iw_2}{w_1 - iw_2}$$

$$= \frac{u_x w_1^2 + v_y w_2^2 + (u_y + v_x) w_1 w_2 + i\{v_x w_1^2 - u_y w_2^2 + (v_y - u_x) w_1 w_2\}}{w_1^2 + w_2^2}$$

$$= \begin{pmatrix} u_x & -v_x \\ v_x & u_x \end{pmatrix}$$ since by the Cauchy-Riemann equations $u_x = v_y$ and $u_y = -v_x$

then $\mathbf{f}'(\mathbf{z}_0) = u_x + iv_x = v_y + iu_y$. Thus,

$$\mathbf{f}'(\mathbf{z}_0) = u_x I + v_x J$$

where $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ [13].

Let $L(B_1, B_2)$ be the Banach space of bounded linear operators from B_1 to B_2 with norm

$$\|L\| = \sup_{x \in B_1, x \neq 0} \frac{\|Lx\|_{B_2}}{\|x\|_{B_1}}$$

If $F : S \subseteq B_1 \rightarrow B_2$ has a Fréchet derivative $F'(x)$ at each x in $N_{\delta(\varepsilon)}(x_0)$ where $N_{\delta(\varepsilon)}(x_0) = \{x: \|x - x_0\| \leq \delta(\varepsilon)\}$ contained in S, then when F' is considered as a mapping,

$$F' : N_{\delta(\varepsilon)}(x_0) \subseteq B_1 \rightarrow L(B_1, B_2)$$

itself has a Fréchet derivative at x_0. If it does, $F''(x_0)$ is denoted as

$$F''(x_0) : B_1 \rightarrow L(B_1, B_2),$$

the *second Fréchet derivative* of F at x_0. Then, $F''(x_0)$ is a bounded linear operator on B_1, and

$$F'(x) = F'(x_0) + F''(x_0)(x - x_0) + G(x, x_0)$$

where $G(x, x_0)$ satisfies

$$\lim_{\|x - x_0\| \rightarrow 0} \{\|G(x, x_0)\|_{B_2} / \|x - x_0\|_{B_1}\} = 0.$$

Section 2.3 Properties of the Fréchet Derivative.

From (2.1.3), as seen in the previous section, take the norms on both sides of the equations and assume that $F : B_1 \to B_2$, then

$$\|F(\tilde{x}) - F(x_0)\| = \|F'(x_0)(\tilde{x} - x_0) + G(\tilde{x}, x_0)\|$$

$$\leq \|\tilde{x} - x_0\| \sup \|F'(\eta)\|$$

$$= k\|\tilde{x} - x_0\|$$

where $k = \sup\|F'(\eta)\|$ and $\eta \in B_1$. Evidently the smaller the maximum of $\|F'(\eta)\|$ on B_1, the more rapidly will the iterates converge to the fixed point $\tilde{x}$. The following theorem shows that the asymptotic rate of convergence (that is near $\tilde{x}$) is determined by the Fréchet derivative at $\tilde{x}$ [17].

Theorem 2.3.1 : Let B_1 be an open subset of the Banach space D, and assume that $F : B_1 \to D$ has a fixed point $\tilde{x}$ in B_1. Suppose that F is Fréchet differentiable at $\tilde{x}$ with $\|F'(\tilde{x})\| < 1$. Then given any ε with $0 < \varepsilon < 1 - \|F'(\tilde{x})\|$, there is an open ball S such that if $x_0 \in S(\tilde{x}, \delta)$, the iterates $x_n = F(x_{n-1})$ $(n \geq 1)$ also lie in $S(\tilde{x}, \delta)$, $\lim_{n\to\infty} x_n = \tilde{x}$, and

$$\|x_n - \tilde{x}\| \leq (\|F'(\tilde{x})\| + \varepsilon)^n \|x_0 - \tilde{x}\|. \qquad (2.3.1)$$

Proof : Pick any ε as above. Then from the definition of the Fréchet derivative, there is a $\delta > 0$ such that for any $x \in S(\tilde{x}, \delta)$,

$$\|F(x) - F(\tilde{x}) - F'(\tilde{x})(x - \tilde{x})\| \leq \varepsilon\|x - \tilde{x}\|.$$

Therefore,

$$\|F(x) - \tilde{x}\| = \|F(x) - F(\tilde{x})\|$$

$$\leq \|F(x) - F(\tilde{x}) - F'(\tilde{x})(x - \tilde{x})\| + \|F'(\tilde{x})(x - \tilde{x})\|$$

$$\leq (\|F'(\tilde{x})\| + \varepsilon)\|x - \tilde{x}\| \leq \delta.$$

That is $F(x) \in S(\tilde{x}, \delta)$. It follows by induction that if $x_0 \in S(\tilde{x}, \delta)$ so does x_n for $n \geq 1$. Replacing x with x_{n-1}, letting $k = \|F'(\tilde{x})\| + \varepsilon$ and using the Mean Value Theorem, the above inequality gives,

$$\|x_n - \tilde{x}\| = \|F(x_{n-1}) - F(\tilde{x})\|$$

$$= \|F'(\xi)\|\|x_{n-1} - \tilde{x}\|$$

$$\leq k\|x_{n-1} - \tilde{x}\|$$

where $\xi \in B_1$. Applying this inequality inductively gives:

$$\|x_n - \tilde{x}\| \leq k\|x_{n-1} - \tilde{x}\| \leq k^2\|x_{n-2} - \tilde{x}\| \leq \ldots\ldots \leq k^n\|x_0 - \tilde{x}\|$$

$$\leq (\|F'(\tilde{x})\| + \varepsilon)^n \|x_0 - \tilde{x}\|$$

Since $k < 1$,

$$\lim_{n\to\infty} \|x_n - \tilde{x}\| \leq \lim_{n\to\infty} k^n\|x_0 - \tilde{x}\| = 0$$

and $\{x_n\}$ converges to $\tilde{x}$.

If this process can be iterated in a single real variable, it is a generalization of Newton's method for solving an equation. Newton's method is derived by assuming that $f(x) = 0$ and is based on Taylor polynomials [14]. Suppose that $f \in C^2[a, b]$. Let $\tilde{x} \in [a, b]$ be an approximation to p such that $f(p) = 0$, $f'(\tilde{x}) \neq 0$ and $|p - \tilde{x}|$ is small. Consider the first Taylor polynomial for f(x) expanded about $\tilde{x}$,

$$f(x) = f(\tilde{x}) + (x - \tilde{x})f'(\tilde{x}) + \frac{(x-\tilde{x})^2}{2} f''(\xi(x)),$$

where $\xi(x)$ lies between x and $\tilde{x}$. Since f(p) = 0, this equation, with x = p, gives

$$0 = f(\tilde{x}) + (p - \tilde{x})f'(\tilde{x}) + \frac{(p-\tilde{x})^2}{2} f''(\xi(p)).$$

Newton's method is derived by assuming that, $|p - \tilde{x}|$ is small, the term involving $(p - \tilde{x})^2$ is negligible and that

$$0 \approx f(\tilde{x}) + (p - \tilde{x})f'(\tilde{x}).$$

Solving for p in this equation gives

$$p \approx \tilde{x} - \frac{f(\tilde{x})}{f'(\tilde{x})}.$$

This sets the stage for the Newton method, which starts with an initial approximation p_0 and generates the sequence $\{p_n\}_{n=0}^{\infty}$, defined by

$$p_n = p_{n-1} - \frac{f(p_{n-1})}{f'(p_{n-1})}, \qquad n \geq 1. \tag{2.3.2}$$

Since the sequence is defined by

$$p_n = g(p_{n-1}), \qquad n \geq 1,$$

then (2.3.2) become

$$g(x) = x - \frac{f(x)}{f'(x)}. \tag{2.3.3}$$

Theorem 2.3.2 : If $g \in C[a, b]$ and $g(x) \in [a, b]$ for all $x \in [a, b]$, then g has a fixed point in [a, b]. Suppose, in addition, that $g'(x)$ exists on (a, b) and that a positive constant $k < 1$ exists with

$$|g'(x)| \leq k < 1, \qquad \text{for all } x \in (a, b),$$

then the fixed point in [a, b] is unique (see proof [14]).

Theorem 2.3.3 : (Fixed-Point Theorem) Let $g \in C[a, b]$ and suppose that $g(x) \in [a, b]$ for all x in [a, b]. Suppose, in addition, that g' exists on (a, b) and that a positive constant $k < 1$ exists with

$$|g'(x)| \leq k < 1, \qquad \text{for all } x \in (a, b).$$

If p_0 is any number in [a, b], then the sequence defined by $p_n = g(p_{n-1})$, $n \geq 1$, converges to the unique fixed point p in [a, b] (see proof [14]).

Definition: Suppose $\{p_n\}_{n=0}^{\infty}$ is sequence that converges to p. If positive constant λ and α exist with

$$\lim_{n \to \infty} \frac{|p_{n+1} - p|}{|p_n - p|^{\alpha}} = \lambda,$$

then $\{p_n\}_{n=0}^{\infty}$ is said to *converge to p of order* α, with asymptotic error constant λ. If $\alpha = 1$, the method is called *linear*.

Theorem 2.3.4 : Let $g \in C[a, b]$ and suppose that $g(x) \in [a, b]$ for all x in [a, b]. Suppose, in addition, that g' exists on (a, b) and that a positive constant $k < 1$ exists with

$$|g'(x)| \leq k < 1, \qquad \text{for all } x \in (a, b).$$

If $g'(p) \neq 0$, then for any number p_0 in [a, b], the sequence

$$p_n = g(p_{n-1}), \qquad n \geq 1,$$

converges only linearly to the unique fixed point p in [a, b].

Proof : Referring back to the Fixed-Point Theorem, the sequence converges to p. Since g' exists on [a, b], apply the Mean Value Theorem to g to show that for any n,

$$p_{n+1} - p = g(p_n) - g(p) = g'(\xi_n)(p_n - p),$$

where ξ_n is between p_n and p. Since $\{p_n\}_{n=0}^{\infty}$ converges to p, $\{\xi_n\}_{n=0}^{\infty}$ also converges to p. Since g' is continuous on [a, b], it follows

$$\lim_{n\to\infty} g'(\xi_n) = g'(p).$$

Thus,

$$\lim_{n\to\infty} \frac{p_{n+1} - p}{p_n - p} = \lim_{n\to\infty} g'(\xi_n) = g'(p) \qquad \text{and} \qquad \lim_{n\to\infty} \frac{|p_{n+1} - p|}{|p_n - p|} = |g'(p)|.$$

Hence, fixed-point iteration exhibits linear convergence if $g'(p) \neq 0$ [14].

CHAPTER 3.

Arnoldi's Method Improved by the Fréchet Derivative

Section 3.1 Why Use Arnoldi's Method ?

Consider solving the linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is an nxn matrix and $\mathbf{b}$ is a nonzero vector. Seldom is it possible to exactly solve a system of linear equations without experimental or round-off errors. Instead the goal of any numerical method is to find the best possible approximations to the exact solution. In a system $\mathbf{Ax} = \mathbf{b}$, the matrix $\mathbf{A}$ and the vector $\mathbf{b}$ may have some experimental errors from measurement such as data errors. Thus the approximate $\mathbf{A}$ and $\mathbf{b}$ are slightly different from the true $\mathbf{A}$ and $\mathbf{b}$. This is called an experimental error. The round-off error is caused by a computer's finite set of floating-point numbers. However, the round-off error is usually much smaller than the experimental error. Therefore, it is important to ask what effect small changes or perturbations in the matrix $\mathbf{A}$ and the vector $\mathbf{b}$ will have on the true solution of a system. This is the question of *sensitivity* of linear systems that is going to be discussed later [7].

Consider a linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is a nonsingular matrix, and $\mathbf{b}$ is a nonzero vector. The system has a unique solution $\mathbf{x}$, which is nonzero. Now suppose a small vector $\delta\mathbf{b}$ is added to $\mathbf{b}$ and consider the perturbed system $\mathbf{A}\tilde{\mathbf{x}} = \mathbf{b} + \delta\mathbf{b}$. This system also has an unique solution $\tilde{\mathbf{x}}$. Hopefully, $\tilde{\mathbf{x}}$ is not too far from $\mathbf{x}$. Let $\tilde{\mathbf{x}} = \mathbf{x} + \delta\mathbf{x}$ where $\delta\mathbf{x}$ denotes the difference between $\tilde{\mathbf{x}}$ and $\mathbf{x}$. Therefore, the equations $\mathbf{Ax} = \mathbf{b}$ and $\mathbf{A}(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b} + \delta\mathbf{b}$ imply that $\mathbf{A}\delta\mathbf{x} = \delta\mathbf{b}$; that is

$$\delta \mathbf{x} = \mathbf{A}^{-1}\delta \mathbf{b}. \tag{3.1.1}$$

Theorem 3.1.1: A vector norm and its induced matrix norm satisfy the inequality

$$\|\mathbf{Ax}\| \le \|\mathbf{A}\|\,\|\mathbf{x}\|$$

for all $\mathbf{A} \in R^{n\times n}$ there exists a nonzero $\mathbf{x} \in R^{n}$ for which equality holds (see proof [7]).

By Theorem 3.1.1 and taking norms on both sides of (3.1.1), it follows

$$\|\delta \mathbf{x}\| \le \|\mathbf{A}^{-1}\|\,\|\delta \mathbf{b}\|. \tag{3.1.2}$$

Similarly the equation $\mathbf{b} = \mathbf{Ax}$ implies that $\|\mathbf{b}\| \le \|\mathbf{A}\|\,\|\mathbf{x}\|$, or equivalently

$$\frac{1}{\|\mathbf{x}\|} \le \|\mathbf{A}\|\frac{1}{\|\mathbf{b}\|}. \tag{3.1.3}$$

Multiplying inequality (3.1.2) and (3.1.3), important inequality is derived

$$\frac{\|\delta \mathbf{x}\|}{\|\mathbf{x}\|} \le \|\mathbf{A}\|\,\|\mathbf{A}^{-1}\|\,\frac{\|\delta \mathbf{b}\|}{\|\mathbf{b}\|} \tag{3.1.4}$$

which provides a bound for $\|\delta \mathbf{x}\|/\|\mathbf{x}\|$ in terms of $\|\delta \mathbf{b}\|/\|\mathbf{b}\|$. The factor $\|\mathbf{A}\|\,\|\mathbf{A}^{-1}\|$ is called the *condition number* of $\mathbf{A}$ and denoted by $\mathbf{k}(\mathbf{A})$. With this new notation (3.1.4) becomes

$$\frac{\|\delta \mathbf{x}\|}{\|\mathbf{x}\|} \le \mathbf{k}(\mathbf{A})\frac{\|\delta \mathbf{b}\|}{\|\mathbf{b}\|} \tag{3.1.5}$$

From (3.1.5), if $\mathbf{k}(\mathbf{A})$ is not too large, then small values of $\|\delta \mathbf{b}\|/\|\mathbf{b}\|$ imply small values of $\|\delta \mathbf{x}\|/\|\mathbf{x}\|$ which is the relative error, i.e., the system is not overly sensitive to perturbations in $\mathbf{b}$. If $\mathbf{k}(\mathbf{A})$ is not too large, $\mathbf{A}$ is said to be *well-conditioned.* By contrast , if $\mathbf{k}(\mathbf{A})$ is large, a small value of $\|\delta \mathbf{b}\|/\|\mathbf{b}\|$ does not guarantee that $\|\delta \mathbf{x}\|/\|\mathbf{x}\|$ will be small. As seen in (3.1.5), there definitely are choices of $\mathbf{b}$ and $\delta \mathbf{b}$ for which the resulting

$\|\delta \mathbf{x}\| / \|\mathbf{x}\|$ is much larger than $\|\delta \mathbf{b}\| / \|\mathbf{b}\|$. In other words the system is potentially very sensitive to perturbations in **b**. Thus if **k(A)** is large, **A** is said to be *ill-conditioned*. Thus the best possible condition number is 1 [7].

The most famous examples of ill-conditioned matrices are Hilbert and Wilkinson matrices. Both Hilbert and Wilkinson matrices are shown in Appendix B. Let matrix **A** be a Hilbert or a Wilkinson matrix and **b** be a vector sum of the columns of matrix **A**. This way the exact solution **x** will be $[1\ 1\ 1.....1]^T$.

Table 3.1.1 and Table 3.1.2 include condition number of different matrices and error comparison between Arnoldi's method and Matlab's "backslash" method on the Hilbert matrix and the Wilkinson matrix. Let $\dot{\mathbf{x}}$ be an approximate solution of Matlab's "backslash" method and $\ddot{\mathbf{x}}$ be an approximate solution of Arnoldi's method.

Table 3.1.1 Hilbert matrix

	Condition number	$\|\mathbf{x} - \dot{\mathbf{x}}\|$	$\|\mathbf{x} - \ddot{\mathbf{x}}\|$
Hilbert matrix$_{15 \times 15}$	3.2878×10^{17}	5.3350	2.9899×10^{-5}
Hilbert matrix$_{30 \times 30}$	2.3121×10^{19}	8.1137	3.8345×10^{-7}
Hilbert matrix$_{50 \times 50}$	8.9532×10^{18}	10.8089	1.5003×10^{-7}

Table 3.1.2 Wilkinson matrix

	Condition number	$\|\mathbf{x} - \dot{\mathbf{x}}\|$	$\|\mathbf{x} - \ddot{\mathbf{x}}\|$
Wilkinson $_{15 \times 15}$	30.5202	20.9462	0.6553
Wilkinson$_{30 \times 30}$	71.6919	50.6757	0.2739
Wilkinson$_{50 \times 50}$	118.71476	80.3343	0.3263

As seen in Table 3.1.1 and Table 3.1.2, Arnoldi's method can solve ill-conditioned problems much more accurately than Matlab's "backslash" method since Arnoldi's method is done by using a projection method with an iterative algorithm. Iterative algorithms are often used for solving linear systems of large dimension, since the time required for sufficient accuracy is an improvement over that required for direct techniques such as the Gaussian elimination method or Matlab's "backslash" method. For large systems with a high percentage of zero entries such as Hilbert matrix, however, iterative techniques are efficient in terms of both computer storage and computational time. Systems of this type arise frequently in circuit analysis and in the numerical solution of boundary-value problems and partial-differential equations [7].

Moreover, Arnoldi's method is an orthogonal projection method onto a Krylov subspace $\mathbf{K}_m$ as defined in Chapter 1. The Arnoldi process can reduce a given large matrix $\mathbf{A}_{n \times n}$ into a smaller matrix $\mathbf{H}_{m \times m}$, $m \leq n$, of upper Hessenberg form. This method achieves a projection process onto the Krylov subspace

$$\mathbf{K}_m = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots\ldots\ldots, \mathbf{A}^{m-1}\mathbf{r}_0\}$$

where $\mathbf{r}_0$ is the initial residual vector. Although the process should theoretically produce the exact solution, in at most n steps, it is well known [6] that a satisfactory accuracy is often achieved for values of m less than n. In general, the treatment of very large problems in linear algebra is done by using projection methods [20].

Section 3.2 Rates of Convergence for Arnoldi's Method. (This section is based on "Krylov Subspace Methods for Solving Large Unsymmetric Linear Systems", by Yousef Saad [22].)

To solve a system of linear equations (1.1) of size n by Arnoldi's method, solve the problem with a restricted subspace $\mathbf{K}_m$, with dimension m such that $m << n$. After computing the approximate solution $\mathbf{x}_m$ with the maximum number of steps allowed, however, the accuracy may still be found unsatisfactory. This naturally raises the question of how to improve the accuracy of $\mathbf{x}_m$ obtained. The simplest method is to restart the algorithm with initial vector $\mathbf{x}_0$ replaced by the approximation $\mathbf{x}_m$ obtained after m iterations. In Chapter one, convergence is achieved in at most n steps where n is the dimension of A. Therefore, the problem is not to show the convergence but rather to establish theoretical error bounds showing that one can obtain satisfactory accuracy. In view of equation (1.6), it is equivalent to study either the convergence of $\mathbf{x}_m$ to $\mathbf{x}^*$ or the convergence of $\mathbf{z}_m$ to $\mathbf{z}^*$ where x^* denotes the exact solution of (1.1) and $\mathbf{z}^*$ denotes the exact solution of (1.5). In addtition, Corollary 1.1.1 of Chapter one shows that the convergence can be studied in terms of $\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\|$, where $\mathbf{Q}_m$ is the orthogonal projection onto the Krylov subspace $\mathbf{K}_m = \text{span}\{\mathbf{r}_0, \mathbf{A}\mathbf{r}_0, \ldots, \mathbf{A}^{m-1}\mathbf{r}_0\}$. Let $\mathbf{P}_k$ be the space of polynomials of degree not exceeding k. Then, a useful expression for the distance $\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\|$ can be derived by remarking that $\mathbf{K}_m$ is nothing but the subspace of R^n constituted by all the elements $q(\mathbf{A})\mathbf{r}_0$ where q belongs to $\mathbf{P}_{m-1}$.

Proposition 3.2.1: The distance $\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\|$ between $\mathbf{z}^*$ and the Krylov subspace $\mathbf{K}_m$

satisfies

$$\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\| = \min_{\substack{p \in p_m \\ p(0)=1}} \|p(\mathbf{A})\mathbf{x}^*\| \tag{3.2.1}$$

where p is any polynomial (see proof [22]).

In order to obtain an upper bound for (3.2.1), we shall assume that **A** admits n eigenvectors $\phi_1, \phi_2, \ldots, \phi_n$ of norm one, associated with the eigenvalues $\lambda_1, \ldots, \lambda_n$. Then the solution $\mathbf{z}^*$ can be expressed as

$$\mathbf{z}^* = \sum_{i=1}^{n} \alpha_i \phi_i,$$

and we can formulate the next theorem.

Theorem 3.2.1: Set $\alpha = \sum_{i=1}^{n} |\alpha_i|$, where the α_i are the components of the solution $\mathbf{z}^*$ in the eigenbasis of **A**, then

$$\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\| \le \alpha \min_{\substack{p \in p_m \\ p(0)=1}} \max_{j=1,\ldots,n} |p(\lambda_j)| \tag{3.2.2}$$

where p is any polynomial (see proof [22]).

Let $\varepsilon^m = \min_{\substack{p \in p_m \\ p(0)=1}} \max_{j=1,\ldots,n} |p(\lambda_j)|$, so (3.2.2) can be simplified to $\|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\| \le \alpha \varepsilon^m$.

According to Corollary 1.1.2 and (3.2.2), the equation (1.6) becomes

$$\|\mathbf{x}_m - \mathbf{x}^*\| = \|\mathbf{z}_m - \mathbf{z}^*\| \le \sqrt{1 + r_m^2 k_m^2}\, \|(\mathbf{I} - \mathbf{Q}_m)\mathbf{z}^*\| \le \alpha \sqrt{1 + r_m^2 k_m^2}\, \varepsilon^m$$

where α is components of $\mathbf{z}^*$ and unfortunately there is no upper bound for α.

ε^m is an error bound and needs to be shown decreasing rapidly to zero. Note that $\varepsilon^n = 0$ which shows that the processes will give the exact solution in at most n steps since

n is dimension of **A**. The rate of convergence of the sequence ε^m to zero provides a bound for the actual rate of convergence. Estimating ε^m is, unfortunately, a difficult problem in general. The number ε^m is the degree of best approximation of the zero function by polynomial of degree m satisfying the constraint p(0)=1, over the set λ_1, λ_2.....λ_n.

From Theorem 2.3.1, it follows that the asymptotic rate of convergence (that is near $\tilde{x}$ which is the approximate solution of (1.1)) is determined by the Fréchet derivative at $\tilde{x}$. Therefore, it would be interesting to know how to compute the Fréchet derivative for Arnoldi's method. This will be discussed in the next section.

Section 3.3 How to Compute the Fréchet Derivative for Arnoldi's Method.

Arnoldi's method uses an iterative technique to solve the linear system $\mathbf{Ax} = \mathbf{b}$. The algorithm starts with an initial approximation $\mathbf{x}_0$ to the solution $\mathbf{x}$, and generates a sequence of vectors $\{\mathbf{x}_k\}$ that converges to $\mathbf{x}$. From Arnoldi's algorithm, matrix $\mathbf{A}$, $\mathbf{x}_0$, $\mathbf{b}$ and an error tolerance ε are given. The algorithm gives output for the real solution vector $\mathbf{x}$. Let Arnoldi's algorithm be represented by a function. Let $\mathbf{x}_0$ be the initial vector of $\mathbf{Ax} = \mathbf{b}$ and $\mathbf{x}_0 + \varepsilon\mathbf{h}$ be the initial vector of the perturbed system $\mathbf{A}(\mathbf{x} + \varepsilon\mathbf{h}) = \mathbf{b}$ where $\mathbf{h}$ is a vector such as $\mathbf{e}_1, \mathbf{e}_2 \ldots\ldots\mathbf{e}_k$ where $\mathbf{e}_1 = \{1, 0, \ldots\ldots 0\}^T$ and $\mathbf{e}_2 = \{0, 1, \ldots, 0\}^T$ and so on (see the Diagram 3.2.1).

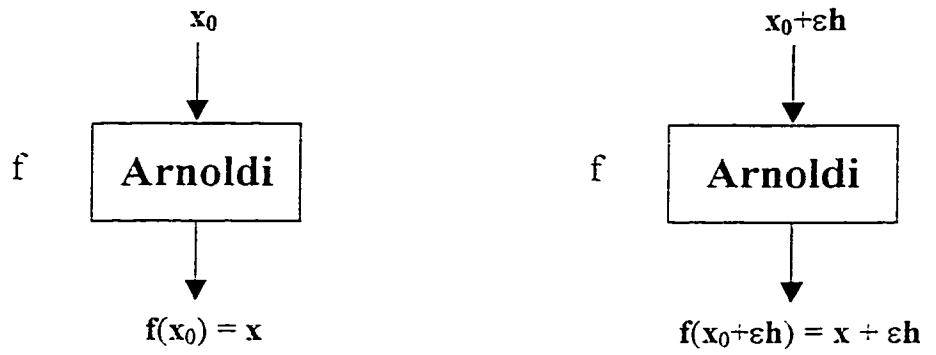

Diagram 3.2.1 Arnoldi's algorithm

By first-order Taylor series and letting $\mathbf{f} : R^n \to R^n$

$$\mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) = \mathbf{f}(\mathbf{x}_0) + \varepsilon \sum_{i=1}^{n} h_i \frac{\partial f_i}{\partial x_i}(x_0) + R_1(\mathbf{h}, \mathbf{x}_0) \qquad (3.3.1)$$

where $R_1(\mathbf{h}, \mathbf{x}_0)$ is the remainder, and in vector form (3.3.1) gives

$$\mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) \approx \mathbf{f}(\mathbf{x}_0) + (\varepsilon\,\mathbf{h} \cdot \nabla)\mathbf{f}(\mathbf{x}_0) \qquad (3.3.2)$$

where

$$\frac{\partial f_i}{\partial x_i}(x_0) = \nabla \mathbf{f}(\mathbf{x_0}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}.$$

This is also called the Jacobian matrix.

Consider an operator f: $V \to U$ where V is a vector space and U is a normed vector space. Let $x \in D(T) \subset V$ and let $x \in V$: if the limit

$$\lim_{\varepsilon \to 0} \frac{f(x + \varepsilon h) - f(x)}{\varepsilon} = df(x)h \tag{3.3.3}$$

exists, and it is called the Gateaux differential of f at x in the direction h, or equivalently, if

$$f(x + \varepsilon h) = f(x) + \varepsilon df(x)h + o(\varepsilon),$$

for $\varepsilon \to 0^+$, where $o(\varepsilon)/\varepsilon \to 0$ as $\varepsilon \to 0^+$. Then the linear operator df(x) is called a gradient or Gateaux derivative. One of the most important applications of the Gateaux derivative is in determining the maxima and minima of a functional. If the limit (3.3.3) is uniform over $\|h\| = 1$, then f is said to have a directional Fréchet derivative at x denoted $f'(x)$.

Theorem 3.3.1: If the functional $f : X \to R$ has a minimum or a maximum at $x \in X$ and df(x) exists, then df(x) = 0 (see proof [15]).

Let $\mathbf{f} : R^n \to R^n$, where **f** is a vector-valued function, then by definition of the Fréchet derivative

$$\|\mathbf{f}(\mathbf{x_0} + \mathbf{s}) - \mathbf{f}(\mathbf{x_0}) - (\mathbf{s} \cdot \nabla)\mathbf{f}(\mathbf{x_0})\| = \|\varepsilon(\mathbf{x_0}, \mathbf{s})\|$$

where $\mathbf{x_0} \in R^n$ and $\mathbf{s} \in R^n$

and

$$\lim_{\|s\|\to 0} \frac{\| \mathbf{f}(\mathbf{x}_0 + \mathbf{s}) - \mathbf{f}(\mathbf{x}_0) - (\mathbf{s}\cdot\nabla)\mathbf{f}(\mathbf{x}_0)\|}{\|\mathbf{s}\|} = 0$$

or putting $\mathbf{s} = \varepsilon\mathbf{h}$ where $\mathbf{h} \in R^n$ and $\|\mathbf{h}\| = 1$, then

$$\lim_{\varepsilon\to 0} \frac{\| \mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) - \mathbf{f}(\mathbf{x}_0) - (\varepsilon\mathbf{h}\cdot\nabla)\,\mathbf{f}(\mathbf{x}_0)\|}{\varepsilon} = 0$$

so that

$$\lim_{\varepsilon\to 0} \frac{\mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) - \mathbf{f}(\mathbf{x}_0)}{\varepsilon} = (\mathbf{h}\cdot\nabla)\mathbf{f}(\mathbf{x}_0)$$

From (3.3.1) and (3.3.2), an algorithm is derived to compute the Fréchet derivative

$$\frac{\mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) - \mathbf{f}(\mathbf{x}_0)}{\varepsilon} = \frac{\mathbf{f}(\mathbf{x}_0) + (\varepsilon\mathbf{h}\cdot\nabla)\,\mathbf{f}(\mathbf{x}_0) - \mathbf{f}(\mathbf{x}_0)}{\varepsilon} = (\mathbf{h}\cdot\nabla)\mathbf{f}(\mathbf{x}_0) \qquad (3.3.4)$$

(see Appendix A).

Example : Given $\mathbf{f}: R^m \to R^n$ and $\mathbf{f}$ be the Fréchet derivative at $\mathbf{x}$ then

$$d\mathbf{f}(\mathbf{x})\mathbf{h} = (\mathbf{h}\cdot\nabla)\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial f_n} \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} \begin{pmatrix} h_1 \\ \cdot \\ \cdot \\ h_n \end{pmatrix}$$

where $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}).....f_m(\mathbf{x}))$ and $\mathbf{x} = (x_1, x_2, ..., x_n)$

Let $\mathbf{f}: R^n \to R^n$ be Fréchet differentiable at $\mathbf{x}$ where $\mathbf{x} = (x_1, x_2....., x_n)$, $\mathbf{f}(\mathbf{x}) = (f_1(x), f_2(x),.....f_n(x))$ and $\mathbf{h} = (h_1, h_2,.....h_n)$, then

$$(\mathbf{h}\cdot\nabla)\mathbf{f} = \sum_{i=1}^{n} h_i \frac{\partial \mathbf{f}}{\partial x_i}(\mathbf{x})$$

$$= \begin{pmatrix} h_1 \frac{\partial f_1}{\partial x_1} + h_2 \frac{\partial f_2}{\partial x_2} + \ldots\ldots\ldots + h_n \frac{\partial f_n}{\partial x_n} \\ \\ \\ \\ \\ h_1 \frac{\partial f_n}{\partial x_1} + h_2 \frac{\partial f_n}{\partial x_2} + \ldots\ldots\ldots + h_n \frac{\partial f_n}{\partial x_n} \end{pmatrix}$$

is an n x1 vector.

If $\mathbf{h} = \mathbf{e}_1$, then $(\mathbf{h} \cdot \nabla)\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} \\ \\ \\ \frac{\partial f_n}{\partial x_1} \end{pmatrix} (\mathbf{x})$. If $\mathbf{h} = \mathbf{e}_2$, then $(\mathbf{h} \cdot \nabla)\mathbf{f}(\mathbf{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_2} \\ \\ \\ \frac{\partial f_n}{\partial x_2} \end{pmatrix} (\mathbf{x}), \ldots\ldots\ldots\ldots$

If $\mathbf{h} = \mathbf{e}_j$, then $(\mathbf{h} \cdot \nabla)\mathbf{f}(\mathbf{x}) = \frac{\partial}{\partial x_j} \begin{pmatrix} f_1 \\ f_2 \\ \\ f_n \end{pmatrix} (\mathbf{x})$ where $j = 3, \ldots\ldots, n$ (3.3.5)

From (3.3.4) and (3.3.5), and take the norms on both sides of equation, assume that $\mathbf{f}: R^n \rightarrow R^n$, then

$$\frac{\| \mathbf{f}(\mathbf{x}_0 + \varepsilon\mathbf{h}) - \mathbf{f}(\mathbf{x}_0) \|}{\|\varepsilon\|} = \| (\mathbf{h} \cdot \nabla)\mathbf{f}(\mathbf{x}_0) \| = \| \frac{\partial}{\partial x_j} \begin{pmatrix} f_1 \\ f_2 \\ \\ f_n \end{pmatrix} (\mathbf{x}_0) \|. \quad (3.3.6)$$

The linear system $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is an nxn matrix and $\mathbf{b}$ is a nonzero vector can be solved by letting $\mathbf{r}(\mathbf{x}) = \mathbf{Ax} - \mathbf{b}$. Since Arnoldi's method is in the iterative form by letting $\mathbf{x}_n = \mathbf{f}(\mathbf{x}_{n-1})$ where n is dimension of matrix $\mathbf{A}$. The process is terminated when the

sequence of vectors $\{\mathbf{x}_n\}$ converges to $\mathbf{x}$. The goal is to find $\mathbf{x}_n$ such that $\mathbf{r}(\mathbf{x}_n) \approx \mathbf{0}$. By fixed-point iteration, given a function $\mathbf{f}$ is defined with a fixed vector $\mathbf{p} = \mathbf{x}_n$ such as

$$\mathbf{f}(\mathbf{x}) = \mathbf{x} - \mathbf{r}(\mathbf{x}) \tag{3.3.7}$$

If the function $\mathbf{f}$ has a fixed vector at $\mathbf{p}$ then the function defined by (3.3.6) has zero at $\mathbf{p}$. Note that the solution of $\mathbf{r}(\mathbf{p}) = \mathbf{0}$ and $\mathbf{p} = \mathbf{f}(\mathbf{p})$ are the same.

In one dimension, it is easy to see by differentiability that if p is a zero of r then $f'(x) = 0$, and it follows from Theorem 2.3.1, it says that $\|f'(\tilde{x})\| < 1$ and $x_n = f(x_{n-1})$, then $\lim_{n\to\infty} x_n = \tilde{x}$. It shows that if the norm of the Fréchet derivative is less than 1, then the sequence $\{x_n\}$ will converge more rapidly to the fixed point $\tilde{x}$. The argument generalizes readily and Theorem 3.3.2 below confirms that the great advantage of speed of convergence for Arnoldi's method is retained in infinite dimensions. The asymptotic rate of convergence is also determined by the Fréchet derivative.

Theorem 3.3.2 : Let V and U be both normed vector spaces and $\mathbf{f} : V \to U$ has a fixed vector $\tilde{x}$ in V. Let V be an open subset of the Banach space D and both U, $V \subseteq R^n$. Suppose that $\mathbf{f}$ is a Fréchet differentiable at $\tilde{x}$ with $\|\mathbf{f}'(\tilde{x})\| < 1$. Then given any

$$0 < \varepsilon < 1 - \|\mathbf{f}'(\tilde{x})\|,$$

there is an open ball S such that if $\mathbf{x}_0 \in S(\tilde{x}, \delta)$, the iterates

$$\mathbf{x}_n = \mathbf{f}(\mathbf{x}_{n-1}) \ (n \geq 1)$$

also lie in $S(\tilde{x}, \delta)$, $\lim_{n\to\infty} \mathbf{x}_n = \tilde{x}$.

Proof : Pick any ε as above. Then from the definition of the Fréchet derivative, there is a

$\delta > 0$ such that for any $\mathbf{x} \in S(\tilde{\mathbf{x}}, \delta)$,

$$\| \mathbf{f}(\mathbf{x}) - \mathbf{f}(\tilde{\mathbf{x}}) - \mathbf{f}'(\tilde{\mathbf{x}})(\mathbf{x} - \tilde{\mathbf{x}}) \| \leq \varepsilon \| \mathbf{x} - \tilde{\mathbf{x}} \|.$$

The result follows from Theorem 2.3.1.

Let $\mathbf{x}$ be the exact solution of (1.1) and $\tilde{\mathbf{x}}$ be the chosen initial guess which is hoped to be the best possible approximation to the exact solution. Theorem 3.3.2 shows that if $\| \mathbf{f}'(\tilde{\mathbf{x}}) \| < 1$, then $\lim_{n \to \infty} \mathbf{x}_n = \tilde{\mathbf{x}}$. The next chapter will use some numerical results of ill-conditioned and/or non-symmetric matrices to support Theorem 3.3.2.

CHAPTER 4.

Data Comparison

Section 4.1 Introduction.

In a practical large-scale computation of the mid-1990s, where iterative algorithms such as Arnoldi's method and Conjugate Gradients method are successful, perhaps a typical result is that they beat direct algorithms such as Gaussian elimination or Householder Triangulation by a factor on the order of 10. As machines get faster and the dimension of the matrix n gets larger in the future, this factor will increase and iterative methods will become more and more important [11].

The most common ways to make Arnoldi's method converge faster are to choose a good initial vector or to use preconditioning techniques. Consider the linear system of equation of (1.1). Preconditioning techniques are to find a matrix **M** such that

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{x} = \mathbf{M}^{-1}\mathbf{b} \tag{4.1.1}$$

where **M** is an $n \times n$ matrix. The system of (4.1.1) should have the same solution as (1.1). If (4.1.1) is solved iteratively, the convergence will depend on the properties of $\mathbf{M}^{-1}\mathbf{A}$ instead of those of **A**. If the *preconditioner* **M** is well chosen, (4.1.1) may be solved much more rapidly than (1.1). For this idea to be useful, it must be possible to compute the operation represented by the product $\mathbf{M}^{-1}\mathbf{A}$ efficiently. This will not mean an explicit construction of the inverse $\mathbf{M}^{-1}$ but the solution of systems of equations of the form

$$\mathbf{M}\mathbf{y} = \mathbf{c}. \tag{4.1.2}$$

If $\mathbf{M} = \mathbf{A}$, then (4.1.2) is the same as (1.1), so applying the preconditioner is as hard as solving the original problem, and nothing has been gained. If $\mathbf{M} = \mathbf{I}$, then (4.1.1) is the same as (1.1), so applying the preconditioner is trivial, and it accomplishes nothing. Between these extremes lie the useful preconditioners, structured enough so that (4.1.2) can be solved quickly, but close enough to A is some sense that an iteration for (4.1.1) converges more quickly than an iteration for (1.1) [3]. What does it mean for M to be "close enough to A?" If $\mathbf{M}^{-1}\mathbf{A}$ is close to the identity matrix, then it is clearly close to a symmetric matrix (the identity), so that Arnoldi's method works better. Usually, when A is indefinite (nonsymmetric, nonpositive), preconditioning techniques with the iterative methods is popular. Saad's [17], [18] papers show how preconditioning techniques work on Arnoldi's method.

From Theorem 3.3.2 of Chapter 3, it shows that if the norm of Fréchet derivative is less than one, then the sequence $\{\mathbf{x}_n\}$ will converge more rapidly to the exact solution **x** of linear equation of (1.1). The goal is to find the best possible approximations to the exact solution **x** of (1.1). It is well known that the numerical results are improved when the size of the Krylov subspace is *larger* and/or the initial vector **x** is *a good guess* [22].

Example: Let A be a Hilbert matrix$_{50 \times 50}$ and **b** be a vector sum of each column of matrix **A**. This way the exact solution **x** will be $[1\ 1.....1]^T$ where is a vector of length 50. Then $\mathbf{b} = \mathbf{Ax} = \mathbf{A} \times [1\ 1.....1]^T$ where $[1\ 1.....1]^T$ is a vector of length 50. Let the initial vector $[0\ 0.....0]^T$ be a vector of length 50. When the size of the subspace is enlarged, Table 4.1.1 shows that numerical results can be improved. Let the size of the subspace be m, $\tilde{\mathbf{x}}$ be the approximate solution, then iterate the algorithm 20 times.

Table 4.1.1 The residual comparison of different subspace

	m (restart) = 7	m (restart) = 9	m (restart)= 10
$\|\mathbf{x} - \tilde{\mathbf{x}}\|$	1.0×10^{-10}	1.0×10^{-14}	1.0×10^{-15}

This chapter will concentrate on using the Fréchet derivative to choose a good initial vector. From Theorem 3.3.2, the goal is to choose an initial vector that has $\sup\|\mathbf{f}'(\tilde{\mathbf{x}})\| < 1$ where $\mathbf{f}$ is Fréchet differentiable at every $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}}$ is an approximate solution of (1.1). Let n = dim(**A**) and $\mathbf{h}_i$ is a vector such that

$\mathbf{h}_1 = \mathbf{e}_1, \mathbf{h}_2 = \mathbf{e}_2, \ldots\ldots, \mathbf{h}_n = \mathbf{e}_n$ where $\mathbf{e}_1 = \{1, 0, \ldots, 0\}^T$ and $\mathbf{e}_2 = \{0, 1, \ldots\ldots, 0)^T$ and so on.

If $\sup\|\mathbf{f}'(\tilde{\mathbf{x}})\| < 1$, then choose the initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_i$ giving the smallest norm of the Fréchet derivative to be the next starting vector will help the rate of convergence of Arnoldi's method (where $\mathbf{x}_0 + \varepsilon\mathbf{h}_i$ is defined at Diagram 3.2.1 and i = 1,....., n). On the other hand, if norms of the Fréchet derivative are greater than 1, then another initial vector needs to be found to make $\|\mathbf{f}'(\tilde{\mathbf{x}})\|$ as small as possible. In the later section, some famous examples of ill-conditioned matrices such as Hilbert and Wilkinson matrices will be examined.

Section 4.2 Hilbert Matrix.

Let **A** be an nxn Hilbert matrix and **b** be a vector sum of each column of matrix **A**. This way the exact solution **x** will be $[1\ 1.....1]^T$ where $[1\ 1.....1]^T$ is a vector of length n. Then $\mathbf{b} = \mathbf{Ax} = \mathbf{A} \times [1\ 1.....1]^T$ where $[1\ 1.....1]^T$ is a vector of length n. Let the initial vector be $[0\ 0.....0]^T$ where $[0\ 0.....0]^T$ is a vector of length n.

Let the norm of the Fréchet derivative be denoted by frenorm and $\mathbf{x}_0 + \varepsilon\mathbf{h}_i$ be an initial vector where is defined in Section 3.3 and i = 1,.....,n. Let n = dim(**A**) and $\mathbf{h}_1 = \mathbf{e}_1$, $\mathbf{h}_2 = \mathbf{e}_2$,...$\mathbf{h}_3 = \mathbf{e}_3$,.....,$\mathbf{h}_n = \mathbf{e}_n$. By Theorem 3.3.2, the goal is to choose an initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_i$ giving the smallest frenorm to be the next starting vector. Hopefully, this initial vector can make Arnoldi's method converge faster. Let Arnoldi/Fréchet derivative be the notation for the algorithm which use the Fréchet derivative to improve Arnoldi's algorithm. Let $\mathbf{x} = \mathbf{b}/\mathbf{A}$ be Matlab's "backslash" method which is Matlab's best solver for the linear system of equations. The following Tables 4.2.1, 4.2.2 and 4.2.3 show the results of the experiment with 3 different algorithms on Hilbert $\text{matrix}_{5 \times 5}$, Hilbert $\text{matrix}_{15 \times 15}$, and Hilbert $\text{matrix}_{30 \times 30}$.

Case one: Let n be 5 and A be a Hilbert $\text{matrix}_{5 \times 5}$.

Let restart be 2 and iteration be 1 where restart is size of the Krylov subspace. Choosing the initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$ to compute the 1st column of frenorm, $\mathbf{x}_0 + \varepsilon\mathbf{h}_2$ to compute the 2nd column of frenorm, etc.

frenorm = $[\underline{0.2065}\ 0.8136\ 0.8443\ 0.8527\ 0.8560]^T$. All frenorms are less than one. The smallest frenorm is 0.2065 (column 1: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$).

Table 4.2.1 (a) Hilbert matrix$_{5 \times 5}$

	x = b/A	Arnoldi	Arnoldi/Fréchet derivative
Initial vector	N/A	$[0\ 0\ 0\ 0\ 0]^T$	$\varepsilon\ [1\ 0\ 0\ 0\ 0]^T$
Residual norm	2.3598	0.0026	3.2458×10^{-4}

Table 4.2.1 (b) Hilbert matrix$_{5 \times 5}$
initial vectors are different columns of A

Arnoldi		Arnoldi / Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	0.0026	column 1 of A + $\varepsilon \mathbf{h}_1$	0.0025
column 2 of A	0.0027	column 2 of A + $\varepsilon \mathbf{h}_1$	0.0016
column 3 of A	0.0026	column 3 of A + $\varepsilon \mathbf{h}_1$	0.0020
column 4 of A	0.0026	column 4 of A + $\varepsilon \mathbf{h}_1$	8.5924×10^{-4}
column 5 of A	0.0025	column 5 of A + $\varepsilon \mathbf{h}_1$	7.1345×10^{-4}

smallest frenorms are all column 1 for different columns of A
all frenorms are less than one

Case two: Let n be 15 and A be a Hilbert matrix$_{15 \times 15}$.

Let restart be 4 and iteration be 1.

frenorm = [0.0456 0.4300 0.8210 0.8566 0.8779 0.9010 0.9213 0.9357 0.9433 0.9442 0.9389 0.9284 0.9135 0.8949 0.8734]^T. All frenorms are less than one. The smallest frenorm is 0.0456 (column 1: initial vector $\mathbf{x}_0 + \varepsilon \mathbf{h}_1$).

Table 4.2.2(a) Hilbert matrix$_{15 \times 15}$

	x = b/A	Arnoldi	Arnoldi/Fréchet derivative
Initial vector	N/A	$[0\ 0\ .....0\ 0\ 0]^T$	$\varepsilon\ [1\ 0.....0\ 0\ 0\ 0]^T$
Residual norm	5.3350	2.9899×10^{-5}	1.1032×10^{-5}

Table 4.2.2(b) Hilbert matrix$_{15 \times 15}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	2.9890×10^{-5}	column 1 of A + $\varepsilon \mathbf{h}_1$	2.9790×10^{-5}
column 3 of A	2.9819×10^{-5}	column 3 of A + $\varepsilon \mathbf{h}_1$	1.7152×10^{-5}
column 9 of A	2.9943×10^{-5}	column 9 of A + $\varepsilon \mathbf{h}_1$	1.3212×10^{-5}

smallest frenorms are all column 1 for different columns of A
all frenorms are less than one

Case three: Let n be 30 and A be a Hilbert matrix$_{30 \times 30}$.

Let restart be 6 and iteration be 1.

frenorm = [0.0106 0.1766 0.6313 0.7959 0.8597 0.8771 0.8911 0.9072 0.9220 0.9331

0.9401 0.9439 0.9456 0.9464 0.9472 0.9486.....0.9639 .....0 9054 0.8827]^T

All frenorms are less than one. Smallest frenorm is 0.0106 (column 1: initial vector $\mathbf{x}_0 + \varepsilon \mathbf{h}_1$).

Table 4.2.3(a) Hilbert matrix$_{30 \times 30}$

	x = b/A	Arnoldi	Arnoldi/Fréchet derivative
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [1 0 0 .....0 0 0]^T
Residual norm	8.1137	3.8345×10^{-7}	1.8622×10^{-7}

Table 4.2.3(b) Hilbert matrix$_{30 \times 30}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	3.8344×10^{-7}	column 1 of A+$\varepsilon\mathbf{h}_1$	3.8190×10^{-7}
column 15 of A	3.8350×10^{-7}	column 15 of A+$\varepsilon\mathbf{h}_1$	1.9933×10^{-7}
column 22 of A	3.8345×10^{-7}	column 22 of A+$\varepsilon\mathbf{h}_1$	1.9526×10^{-7}

smallest frenorms are all column 1 for different columns of A
all frenorms are less than one

These cases show that Arnoldi's method works much better than Matlab's "backslash" method, which is Matlab's best solver. If a machine's zero is 10^{-16}, then it is a well known fact that any direct solver like Gaussian elimination or Matlab's "backslash" method can not deal with Hilbert matrices of order more than 8×8 [4] (see Tables 4.2.1(a), 4.2.2(a), 4.2.3(a)). Moreover, if all frenorms are less than one, then choose the initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_i$ giving the smallest norm of the Fréchet derivative to be the next starting vector, and the above cases show the Arnoldi/Fréchet derivative algorithm converges faster than Arnoldi's algorithm. These data comparison also verify Theorem 3.3.2.

Section 4.3 Wilkinson Matrix.

Let **A** be an nxn Wilkinson matrix and **b** be a vector sum of each column of matrix **A**. This way the exact solution **x** will be $[1\ 1.....1]^T$ where $[1\ 1.....1]^T$ is a vector of length n. The following Tables 4.3.1, 4.3.2, 4.3.3, and 4.3.4 show the results of the experiment with 3 different algorithms on Wilkinson $matrix_{15\times 15}$, Wilkinson $matrix_{20\times 20}$, Wilkinson $matrix_{30\times 30}$, and Wilkinson $matrix_{40\times 40}$. Let $\mathbf{h}_1 = \mathbf{e}_1$, $\mathbf{h}_2 = \mathbf{e}_2$, ... $\mathbf{h}_3 = \mathbf{e}_3$,.....,$\mathbf{h}_n = \mathbf{e}_n$ where n = dim(**A**)

Case one: Let n be 15 and A be a Wilkinson $matrix_{15\times 15}$.

Let restart be 4 and iteration be 1.

Choosing the initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$ to compute the 1st column of frenorm, $\mathbf{x} + \varepsilon\mathbf{h}_2$ to compute the 2nd column of frenorm, etc.

frenorm = [0.0199 0.0252 0.0405 0.0520 0.2079 0.6959 2.0134 3.2866 2.0134 0.6959 0.2079 0.0520 .0.0405 0.0252 0.0199]^T

Most of frenorms are less than one. The smallest frenorm is 0.0119 (column 1: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$).

Table 4.3.1(a) Wilkinson $matrix_{15\times 15}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	$[0\ 0.....0\ 0\ 0\ 0\ 0\ 0]^T$	$\varepsilon\ [1\ 0\ 0\ .....0\ 0\ 0\ ]^T$
Residual norm	20.9462	0.6553	0.6391

Table 4.3.1(b) Wilkinson matrix$_{15 \times 15}$
initial vectors are different columns of A

Arnoldi		Arnoldi / Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 5 of A	0.3855	column 5 of A+$\varepsilon\mathbf{h}_1$	0.3429
column 13 of A	0.5586	column 13 of A+$\varepsilon\mathbf{h}_1$	0.5245
column 15 of A	0.5763	column 15 of A+$\varepsilon\mathbf{h}_1$	0.5031

smallest frenorms are all column 1 for different columns of A
most of frenorms are less than 1

Case two: Let n be 20 and A be a Wilkinson matrix$_{20 \times 20}$.

Let restart be 5 and iteration be 1.

frenorm = [0.0035 0.0052 0.0059 0.0075 0.0087 0.0297 0.1072 0.3302 0.8262 1.5585

1.5585 0.8262 0.3302 0.1072 0.0297 0.0087 0.0075 0.0059 0.0052 0.0035]^T

Most of frenorms are less than one. The smallest frenorm is 0.0035 (column 1: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$).

Table 4.3.2(a) Wilkinson matrix$_{20 \times 20}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [1 0 0 .....0 0 0]^T
Residual norm	30.7901	0.1774	0.1746

Table 4.3.2(b) Wilkinson matrix$_{20 \times 20}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	0.1778	column 1 of A+$\varepsilon\mathbf{h}_1$	0.1753
column 10 of A	0.2633	column 10 of A+$\varepsilon\mathbf{h}_1$	0.2529
column 15 of A	0.5342	column 15 of A+$\varepsilon\mathbf{h}_1$	0.5231

smallest frenorms are all column 1 for different columns of A
most of frenorms are less than 1

Case three: Let n be 30 and A be a Wilkinson matrix$_{30 \times 30}$.

Let restart be 6 and iteration be 1.

frenorm = [0.0035 0.0041 0.0047 0.0053 0.0052 0.0039 0.0060 0.0070 0.0063 0.0129 0.0340 0.1142 0.3334 0.8004 1.4804 1.4805 0.8004 0.3334 0.1142 0.0340 0.0129 0.0063 0.0070 0.0060 0.0039 0.0052 0.0053 0.0047 0.0041 0.0035]^T

Most of frenorms are less than one. The smallest frenorm is 0.0035 (column 1: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$).

Table 4.3.3(a) Wilkinson matrix$_{30 \times 30}$

	x = b/A	Arnoldi	Arnoldi/Fréchet derivative
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [1 0 0 .....0 0 0]^T
Residual norm	50.6757	0.2739	0.2724

Table 4.3.2(b) Wilkinson matrix$_{30 \times 30}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	0.2880	column 1 of A+$\varepsilon\mathbf{h}_1$	0.2863
column 10 of A	0.2720	column 10 of A+$\varepsilon\mathbf{h}_1$	0.2474
column 20 of A	0.3652	column 20 of A+$\varepsilon\mathbf{h}_1$	0.3557

smallest frenorms are all column 1 for different columns of A
most of frenorms are less than 1

Case four: Let n be 40 and A be a Wilkinson matrix$_{40 \times 40}$.

Let restart be 7 and iteration be 1.

frenorm =[0.0030 0.0032 0.0039 0.0036 0.0040 0.0031 0.0030 0.0043 0.0042 0.0037 0.0048 0.0064 0.0063 0.0067 0.0143 0.0356 0.1168 0.3354 0.7972 1.4688 1.4688

0.7982 0.3354 0.1168 0.0356 0.0143 0.0067 0.0063 0.0064 0.0048 0.0037 0.0042

0.0043 0.0030 0..0031 0.0040 0.0036 0.0039 0.0032 0.0030]^T

Most of frenorms are less than one. The smallest of frenorm is 0.0030 (column 1: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_1$).

Table 4.3.4(a) Wilkinson matrix$_{40 \times 40}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [1 0 0 .....0 0 0]^T
Residual norm	80.3343	0.3263	0.3252

Table 4.3.2(b) Wilkinson matrix$_{40 \times 40}$
initial vectors are different columns of A

Arnoldi		**Arnoldi / Fréchet derivative**	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	0.3470	column 1 of A+$\varepsilon\mathbf{h}_1$	0.3457
column 10 of A	0.4740	column10 of A+$\varepsilon\mathbf{h}_1$	0.3181
column 20 of A	0.8453	column 20 of A+$\varepsilon\mathbf{h}_1$	0.8457

smallest frenorms are all column 1 for different columns of A
most of frenorms are less than 1

These cases show that Arnoldi/Fréchet derivative algorithm converges a little bit faster than Arnoldi's algorithm. The reason that the Arnoldi/Fréchet derivative algorithm does not converge rapidly is that not all frenorms are less than one. Theorem 3.3.2 says that the Arnoldi/Fréchet derivative algorithm will help the rate of convergence when frenorms are all less than one.

Section 4.4 Toeplitz Matrix and Certain Random Matrices.

Let **A** be an nxn Toeplitz matrix or a random matrix and **b** be a vector sum of each column of matrix **A**. This way the exact **x** will be $[1\ 1.....1]^T$ where $[1\ 1.....1]^T$ is a vector of length n. The following Tables 4.4.1, and 4.4.2 show the results of the experiment with 3 different algorithms on Toeplitz $matrix_{20\times 20}$ and Toeplitz $matrix_{30\times 30}$. Tables 4.4.3 and 4.4.4 show the experiment results with 3 different algorithms on random $matrix_{20\times 20}$, and random $matrix_{30\times 30}$. Let $\mathbf{h}_1 = \mathbf{e}_1$, $\mathbf{h}_2 = \mathbf{e}_2$,..... $\mathbf{h}_3 = \mathbf{e}_3$,....., $\mathbf{h}_{40} = \mathbf{e}_{40}$. Let n be dimesion of **A**.

Case one: Let n be 20 and A be a Toeplitz $matrix_{20\times 20}$.

Let restart be 4 and iteration be 1.

frenorm = [3.4087 9.0196 9.5816 3.5703 3.4564 3.1382 2.9633 2.8586 3.0981 2.7877 2.5978 2.5587 2.4939 2.5703 2.5783 2.8343 1.3124 1.2740 1.1564 0.6752]^T

Almost all of frenorms are not less than one. The smallest frenorm is 0.6752 (column 20: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_{20}$).

Table 4.4.1(a) Toeplitz $matrix_{20\times 20}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	$[0\ 0.....0\ 0\ 0\ 0\ 0\ 0]^T$	$\varepsilon\ [0\ 0\ 0\ .....0\ 0\ 1\ ]^T$
Residual norm	24.6118	2.8510	2.2107

Table 4.4.1(b) Toeplitz matrix$_{20 \times 20}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 3 of A	2.8774	column 3 of A+$\varepsilon \mathbf{h}_{19}$	5.2747
column 10 of A	3.5641	column 10 of A+$\varepsilon \mathbf{h}_{18}$	4.3593
column 20 of A	1.5761	column 20 of A+$\varepsilon \mathbf{h}_{19}$	1.8417

smallest frenorms are column 19 for column 3 and column 20 of A
smallest frenorm is column for column 10 of A
most of frenorms are greater than 1

Case two: Let n be 30 and A be a Toeplitz matrix$_{30 \times 30}$.

Let restart be 6 and iteration be 1.

frenorm = [0.6941 0.7443 0.7630 1.4816 1.0198 2.2459 2.2120 2.2248 2.1883 2.2049 2.2186 2.4179 2.2275 2.2091 2.2091 2.2091 2.2091 2.2275 2.5899 .2.3198 2.2888 2.3982 2.8555 3.4412 4.4876 3.2418 3.0381 3.4083 8.5818 5.9610]^T

Most of frenorms are not less than 1. The smallest frenorm is 0.6941(column 1: initial vector $\mathbf{x}_0 + \varepsilon \mathbf{h}_1$).

Table 4.4.2(a) Toeplitz matrix$_{30 \times 30}$

	x = b/A	Arnoldi	Arnoldi/Fréchet derivative
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [1 0 0 .....0 0 0]^T
Residual norm	5.5426	0.5095	0.6583

Table 4.4.2(b) Toeplitz matrix$_{30 \times 30}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 1 of A	0.6487	column 1 of A+$\varepsilon\mathbf{h}_3$	1.8704
column 10 of A	1.8229	column 10 of A+$\varepsilon\mathbf{h}_3$	1.7456
column 29 of A	1.1911	column 29 of A+$\varepsilon\mathbf{h}_3$	0.9174

smallest frenorms are all column 3 for different columns of A
most of frenorms are greater than 1

Case three: Let n be 20 and A be a random matrix$_{20 \times 20}$.

Let restart be 4 and iteration be 1.

frenorm = [3.7810 13.4305 2.7054 8.6749 2.1543 2.8550 7.0237 13.0593 9.3421 4.8687 2.3613 5.0654 13.5994 11.6068 2.7702 6.3118 3.6759 8.6686 5.9674 11.2330]^T

None of frenorms are less than one. The smallest frenorm is 2.1543(column 5: initial vector $\mathbf{x}_0 + \varepsilon\mathbf{h}_5$).

Table 4.4.3(a) Random matrix$_{20 \times 20}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [0 0 0 0 1 0.....0]^T
Residual norm	43.3501	1.3353	20.1290

Table 4.4.3(b) Random matrix$_{20 \times 20}$
initial vectors are different columns of A

Arnoldi		Arnoldi/Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
column 5 of A	5.7818	column 5 of A+$\varepsilon \mathbf{h}_{17}$	9.7079
column 13 of A	5.8961	column 13 of A+$\varepsilon \mathbf{h}_{10}$	10.9935
column 20 of A	5.3958	column 20 of A+$\varepsilon \mathbf{h}_{12}$	10.9385

smallest frenorm is column 17 for column 5 of A
smallest frenorm is column 10 for column 13 of A
smallest frenorm is column 12 for column 20 of A
all frenorms are greater than one

Case four: Let n be 30 and A be a random matrix$_{30 \times 30}$.

Let restart be 6 and iteration be 1

frenorm = [14.5295 20.7079 10.5300 27.2156 61.6342 19.8325 38.6369 31.3958 33.9151 13.3706 12.0396 23.4363 45.2501 32.4776 7.4924 29.8951 10.5123 18.3674 16.4430 22.7575 28.8193 32.6818 36.1187 10.2872 54.0490 27.3549 35.1645 24.4010 42.7500 8.1658]^T

None of frenorms are less than one. The smallest frenorm is 7.4924(column 15: initial vector $\mathbf{x}_0 + \varepsilon \mathbf{h}_{15}$).

Table 4.4.4(a) Random matrix$_{30 \times 30}$

	x = b/A	**Arnoldi**	**Arnoldi/Fréchet derivative**
Initial vector	N/A	[0 0.....0 0 0 0 0 0]^T	ε [0.....0 1 0.....0]^T
Residual norm	79.0706	5.1023	7.6081

Table 4.4.4(b) Random matrix$_{30 \times 30}$
initial vectors are different columns of A

Arnoldi		Arnoldi / Fréchet derivative	
Initial vector	**Residual norm**	**Initial vector**	**Residual norm**
Column 1 of A	13.1185	column 1 of A+$\varepsilon \mathbf{h}_{25}$	27.0935
Column 15 of A	14.3836	column 15 of A+$\varepsilon \mathbf{h}_{11}$	29.8729
Column 25 of A	17.0254	column 25 of A+$\varepsilon \mathbf{h}_{25}$	20.0099

smallest frenorms are column 25 for column 1 and 25 of A
smallest frenorms is 11 for column 15 of A
all frenorms are greater than one

These cases show that the Arnoldi/Fréchet derivative algorithm does not help the rate of convergence for Arnoldi's algorithm because all frenorms are greater than one by all initial vectors (see Tables 4.4.1, 4.4.2, 4.4.3, 4.4.4). Theorem 3.3.2 says that when norm of the Fréchet derivative is less than one, the Arnoldi/Fréchet derivative algorithm converges more rapidly than Arnoldi's algorithm (see Tables 4.2.1, 4.2.2, 4.2.3). Conversely, if norm of the Fréchet derivative is not less than one, then the Arnoldi/Fréchet derivative algorithm does not improve Arnoldi's method (see Tables 4.4.1, 4.4.2, 4.4.3, 4.4.4). Therefore, this chapter shows that the Arnoldi/Fréchet derivative algorithm helps the rate of convergence for general $n \times n$ Hilbert and Wilkinson matrices but not Toeplitz and certain random matrices.

CHAPTER 5

Conclusions and Future Work

Arnoldi's method constructs an orthonormal basis of subspace called the Krylov subspace. Based on the Arnoldi process, this thesis discusses how the choice of the initial vector $\mathbf{x}_0$ plays an important role. This thesis has examined how the Fréchet derivative can improve the convergence of Arnoldi's method by choosing the best possible initial vector $\mathbf{x}_0$ when it is applied to ill-conditioned and/or non-symmetric matrices.

In Chapter 4, this thesis has given several comparisons between the residual norms for Matlab's "backslash" method, Arnoldi's, and Arnoldi/Fréchet derivative's method. If all norms of Fréchet derivative are less than 1, the Arnoldi/Fréchet derivative process converges faster than the Arnoldi process. The Arnoldi/Fréchet derivative process is found to be effective on ill-conditioned matrices such as Hilbert and Wilkinson matrices. These numerical results verify Theorem 3.3.2 is true. On the other hand, if norms of Fréchet derivatives are greater than 1, the Arnoldi/Fréchet derivative process does not help the rate of convergence. Arnoldi/Fréchet derivative's method fails to converge to the exact solution when the method is applied to non-symmetric matrices such as random and Toeplitz matrices. If A is a non-symmetric positive real matrix, preconditioning techniques are popular in conjunction with iterative methods such as Arnoldi's and Conjugate Gradient methods; see [21, 22]. These papers modify Arnoldi's method by using preconditioning techniques and show some numerical results of residual norms.

By theorem 3.3.2, it would be interesting to research making $(\mathbf{h} \cdot \nabla)\mathbf{f}$ as small as possible because if the norm of the Fréchet derivative is less than one, the Arnoldi/Fréchet derivative algorithm converges more rapidly than Arnoldi's algorithm. A challenging problem is to determine what direction of $\mathbf{h}$ will give the smallest norm of the Fréchet derivative. Also it will be interesting to study solving the system of linear equations of any matrix $\mathbf{A}$ by other techniques such as triangular factorization, an incomplete LU decomposition, or Generalized Minimum Residual methods, and then trying to accelerate them by the Fréchet derivative.

APPENDIX A

PROGRAM LISTING

I. Arnoldi's Method

```
function[x,r,error,flag]=fom(A, x, b, restrt, tol, iter)
%*********************************************************************
%This program creates an H, then solves for H*y=beta*e1,& x=x+V*y
%Only one iteration is performed, Use "up arrows" to do next iter.
%*********************************************************************
%       -- Iterative template routine --
%       August 3, 1994          By Dr. Saleem
%       Details of the algorithm are described in Saad's Notes
%
%[x,error,iter, flag,H]=fom(A,x,b,restrt,tol,iter)
%
%fom.m sloves the linear system Ax=b
%using Arnoldi's method with restarts.
%
%input
%       A       REAL nonsymmetric positive definite matrix
%       x       REAL initial guess vector
%       b       REAL right hand side vector
%       restrt  The max dimension of matrix H
%       tol     REAL error tolerance
%
%output
%       x       REAL solution vector
%       error   REAL error norm
%       H       Hessenberg matrix, with the extra row
%       norm    norm of REAl solution vector

iter=iter+1;
        bnrm2=norm(b);
        if (bnrm2==0.0), bnrm2=1.0; end

        r=(b-A*x);
        error=norm(r)/bnrm2;
        if (error < tol)
         flag='initial guess was very accurate.Arnoldi-loop by -passed'
        end
```

```
% Initialize workspace
[n,n]= size(A);
m=restrt;
V(1:n,1:m+1)=zeros(n,m+1);
H(1:m+1,1:m)=zeros(m+1,m);
e1=eye(n,1);

% Construct Orthonormal basis using Gram-Schmidt
r=(b-A*x);
V(:,1)=r/norm(r);
s=norm(r)*e1;
for i=1:m,
        w=(A*V(:,i));
        for k=1:i,
                H(k,i)=w'*V(:,k);
                w=w-H(k,i)*V(:,k);
        end
        H(i+1,i)=norm(w);
        V(:,i+1)=w/H(i+1,i);
end
        y=inv(H(1:m,1:m))*s(1:i);
        x=x+V(:,1:m)*y;
        r=(b-A*x);
        error=norm(b-A*x);
```

II. Arnoldi's Method improved by the Fréchet derivative

```
% This program calculates Fréchet derivative in order to improve Arnoldi algorithm. %
MATLAB 4.0 for MS-Windows was used for this program.
%
clear
%
% Set up initial data
% matrix A, restrt, and initial vectors change
A=hilb(5,5);
x=zeros(5,1);              % initial vector x0
b=sum(A')';
restrt=2;
tol=0.01;
iter=1;
%
% Call Arnoldi's algorithm
[xold,r,error,flag]=fom(A,x,restrt,tol,iter);     % f(x0)
% output the norms
error
%
% Calculate the Fréchet derivative
% i is dimension of A
for i=1:5
htmp = zeros(5,1);
htmp(i) = 1;
epi = 0.01;
xtol=x + (epi * htmp);          % initial vector x0 + εh
% Call Arnoldi's algorithm
[xnew(:,i),r(:,i),error(:,i),flag]=fom(A,stol,b,restrt,tol,ter);    % f(x0 + εh)
deriv(:i)=(xnew(:,i)-xold)/epi;
frechet(i)=norm(deriv(:,i));
%
% output the norms of the Fréchet derivative
frechet
```

APPENDIX B

A matrix $\mathbf{H_m} = (h_{ij})$ is called an *upper Hessenberg* matrix if $h_{ij} = 0$ whenever $i > j + 1$. Thus an upper Hessenberg matrix has the form

$$\mathbf{H_m} = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & : & : & : & : & h_{1,m-2} & h_{1,m-1} & h_{1,m} \\ h_{2,1} & h_{2,2} & h_{2,3} & : & : & : & : & h_{2,m-2} & h_{2,m-1} & h_{2,m} \\ 0 & h_{3,2} & h_{3,3} & : & : & : & : & h_{3,m-2} & h_{3,m-1} & h_{3,m} \\ 0 & 0 & h_{4,3} & : & : & : & : & : & : & : \\ : & 0 & 0 & : & : & : & : & : & : & : \\ : & : & : & : & : & : & : & : & : & : \\ : & : & : & : & : & : & : & : & : & : \\ 0 & : & : & : & : & : & : & h_{m-2,m-2} & h_{m-2,m-1} & h_{m-2,m} \\ 0 & 0 & : & : & : & : & : & h_{m-1,m-2} & h_{m-1,m-1} & h_{m-1,m} \\ 0 & 0 & 0 & : & : & : & : & 0 & h_{m,m-1} & h_{m,m} \end{bmatrix}$$

$$\tilde{\mathbf{H}}_m = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & : & : & : & : & h_{1,m-2} & h_{1,m-1} & h_{1,m} \\ h_{2,1} & h_{2,2} & h_{2,3} & : & : & : & : & h_{2,m-2} & h_{2,m-1} & h_{2,m} \\ 0 & h_{3,2} & h_{3,3} & : & : & : & : & h_{3,m-2} & h_{3,m-1} & h_{3,m} \\ 0 & 0 & h_{4,3} & : & : & : & : & h_{4,m-2} & h_{4,m-1} & h_{4,m} \\ : & 0 & 0 & : & : & : & : & : & : & : \\ : & : & 0 & : & : & : & : & : & : & : \\ : & : & : & : & : & : & : & : & : & : \\ 0 & : & : & : & : & : & : & : & : & : \\ 0 & 0 & : & : & : & : & : & h_{m-1,m-2} & h_{m-1,m-1} & h_{m-1,m} \\ 0 & 0 & 0 & : & : & : & : & 0 & h_{m,m-1} & h_{m,m} \\ 0 & 0 & 0 & : & : & : & : & 0 & 0 & h_{m+1,m} \end{bmatrix}$$

The $n \times n$ Hilbert matrix $\hat{\mathbf{H}}_n$ is defined by $\hat{\mathbf{H}}_n(i,j) = \dfrac{1}{i+j-1}$ where $1 \le i$, and $j = 1, \ldots\ldots, n$. Thus the $n \times n$ Hilbert matrix has a form

$$\hat{\mathbf{H}}_n = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots\cdots\cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots\cdots\cdots & \frac{1}{n+1} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \frac{1}{n} & \frac{1}{n+1} & \cdots\cdots\cdots & \frac{1}{i+j-2} & \frac{1}{i+j-1} \end{bmatrix}$$

where $1 \le i$, and $j = 1, \ldots\ldots, n$.

The $n \times n$ Wilkinson matrix $\mathbf{W}_n$ is defined by

$$\mathbf{W}_n = \begin{bmatrix} a_1 & b & 0 & 0 & : & : & 0 & 0 \\ b & a_2 & b & 0 & : & : & : & : \\ 0 & b & a_3 & b & : & : & : & : \\ : & 0 & b & a_4 & : & : & : & : \\ : & : & 0 & b & : & : & 0 & : \\ : & : & : & : & : & : & b & 0 \\ : & : & : & : & : & b & a_{n-1} & b \\ 0 & 0 & 0 & 0 & : & 0 & b & a_n \end{bmatrix}$$

where $a_1, a_2, \ldots\ldots, a_n$, are any real number but not all of them are zeros, and b is any real number, but not zero.

REFERENCES

1. A. Cheer and M. Saleem, Acceleration of Convergence by Shifting the Spectrum of Implicit Finite Difference Operators Associated with the Equations of Gas Dynamics, *International Journal for Numerical Methods in Fluids*, 12, (1991), 443-462.

2. Arthur Wouk, *A Course of Applied Functional Analysis*, A Wiley-Interscience Publication, New York, (1979), 263-270.

3. Bruaset Magnus, *A survey of preconditioned iterative methods, Pitman Reasearch notes in Mathematics Series*, Longman Scienticfic & Technical, San Francisoco, 1995.

4. Carl Jagels and Lothar Reichel, The isometric Arnoldi process and an Application to Iterative Solution of Large Linear Systems, *Iterative Methods in Linear Algebra*, R. Beuwens and P. de Groen, ed., Elsevier Science Publishers B.V., Armsterdam, (1992), 361-369.

5. Cong Wen Xiang, The Iterative Method and Its Convergence for Solving Coefficient Inverse Problem of Two Dimension wave Equation, *Math. Appl.* 8, (4), (1995), 342-354.

6. David Bau and N. Lloyd Trefethen, *Numerical Linear Algebra.*, Society for Industrial and Applied Math, Philidelphia, 1997.

7. David S. Watkins, *Fundamentals for Matrix Computations*, John Wiley & Sons, Inc., New York, 1991.

8. Dominikus Noll, 'Directional differentiability of the Metric Projection in Hilbert Space', *Pacific Journal of Math*, 170, (2), (1995), 241-268.

9. Jerrod E. Marsden and Anthony J. Tromba, *Vector Calculus*, W. H. Freeman and Company, New York, 1988.

10. Marlis Hochbruck, Christian Lubich, and Hubert Selhofer, 'Exponential Integrators for Large Systems of Differential Equations', SIAM J. Sci. Comput., 19, (5), (1998), 1552-1574.

11. Peter N. Brown, A Theoretical Comparison of the Arnoldi and GMRES alogrithms, *SIAM J. Sci. Stat. Comput.*, 12, (1), (1991), 58-78.

12. R. D. Milne, *Applied Functional Analysis An Introductory Treatment*, Pitman Advanced Publishing Program, San Francisco, 1980.

13. R. E. Moore, *Computational Functional Analysis*, Ellis Horwwd Limited, New York, 1985.

14. R. L. Burden and J. D. Faires, *Numerical Analysis.*, PWS Publishing Company, Boston, 1985.

15. Ruth F. Curtain and A. J. Pritchard, *Functional Analysis in Modern Applied Mathematics*, Academic Press., New York, 1977.

16. T. M. Flett, *Differential Analysis*, Cambridge University Press, Boston, 1980.

17. V. Hutson and J. S. Phm, *Applications of Functional Analysis and Operator Theory*, Academic Press, New York, 1980.

18. W. E. Arnoldi, 'The principle of Minimized Iterations in the Solution of the Matrix Eignvalue Problem', *Quart. Appl. Math.*, 9, (1951), 17-29.

19. Xiao J. Zhong, Refined Iterative Algorithms Based on Arnoldi's Process for Large Unsymmetric Eigenproblems, *Linear Algebra App.*, 259, (1997), 1-23.

20. Yousef Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

21. Yoursef Saad, 'Practical use of Some Krylov subspace Methods for Solving indefinite and Nonsymmetric Linear Systems', *SIAM J. Sci. Stat. Compt.*, 5, (1), (1984), 203-228.

22. Yousef Saad, 'Krylov Subspace Methods for Solving Large Unsymmetric Linear Systems', *Math. Comp.*, 37, (1981), 105-126.

23. Matlab Version 5 - Computer Software by MATHWORKS, (1996).